Funktionales Clustern von Transaktionsverläufen

Birgit Oellinger

Funktionales Clustern von Transaktionsverläufen

Master Thesis

verfasst von

Birgit Oellinger

Gutachter:

Prof. Dr. Friedrich Leisch

AG Computationale Statistik

Institut für Statistik

Ludwig–Maximilians–Universität München

25. Januar 2010

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

KAPITEL 1

Einführung

Besonders in der Marketingpraxis werden Analysten häufig mit der Frage nach einer geeigneten Segmentierung von Kunden anhand ihrer Transaktionsverläufe, der Anzahl ihrer Einkäufe zu aufeinanderfolgenden Zeitpunkten, konfrontiert. Ein Blick auf diese Schar an Kurven lässt schnell erahnen, dass eine Einteilung von Käufern mit bloßem Auge nicht möglich und auch nicht sinnvoll ist. Die Lösung solcher Segmentierungsprobleme stellt beispielsweise die *Clusteranalyse* dar. Clusteralgorithmen werden i.d.R. dazu verwendet, ähnliche Gruppen innerhalb einer Menge von Beobachtungen aufzudecken. Bei funktionalen Daten dient die Clusteranalyse demselben Zweck. Ihr Ziel ist es, repräsentative Kurvenverläufe zu finden, die bestmöglich die Variationen in den Daten beschreiben.
Im Bereich der funktionalen Clusteranalyse existieren mehrere unterschiedliche Ansätze, die in ihrer Ursprungsform mehr oder minder gut geeignet erscheinen, Transaktionsverläufe zu clustern. So sollen zunächst verschiedene Möglichkeiten der funktionalen Clusteranalyse als Teilgebiet der *funktionalen Datenanalyse* vorgestellt werden und anschließend ein bestimmtes Verfahren so modifiziert werden, dass die Vorgehensweise besser mit den speziell betrachteten Verläufen abgestimmt ist. Abschließend erfolgt eine Analyse von realen Transaktionsverläufen unter Verwendung verschiedener heuristischer Clustermethoden. Dabei richtet sich der Fokus auf neue Möglichkeiten Transaktionsverläufe angemessen zu partitionieren.

KAPITEL 2

Funktionale Daten

Das Interesse an *funktionaler Datenanalyse*[1] entspringt den verschiedensten Problemstellungen und Themenbereichen. Nützlich ist sie in Anwendungen, bei denen diskret gemessenen Werten ein funktionaler Verlauf unterstellt werden kann. So ist oft eine Veränderung über die Zeit ein stetiger Prozess und einzelne Messwerte sind nur Momentaufnahmen zu bestimmten Zeitpunkten. Solche zeitlichen Veränderungen lassen sich beispielsweise bei einem Wachstumsprozess beobachten. Eine Veränderung von einem Messwert zum nächsten erfolgt hier offensichtlich nicht sprunghaft, sondern fortlaufend über die Zeit.
Neben Wachstumskurven stellen auch unter anderem ökonometrische Daten, meteorologische Messungen, Genexpressionsdaten, Reaktionszeiten oder Preisentwicklungen funktionale Verläufe dar. In der Praxis liegen solche Daten univariat oder multivariat als Vektor von Beobachtungen in diskreter Zeit vor, entstehen aber eigentlich als Funktion über die Zeit t oder allgemeiner als Funktion einer stetigen Variablen. Eine Funktion x wird somit nicht für jeden Wert von t erfasst, sondern vielmehr werden funktionale Daten normalerweise als T Paare (t_j, y_j), $j = 1, \ldots, T$, beobachtet und aufgezeichnet, wobei mit y_j der möglicherweise mit einem Fehler ϵ_j behaftete Messwert zum Zeitpunkt t_j bezeichnet wird. Das *signal-to-noise*-Modell sieht folgendermaßen aus:

$$y_j = x(t_j) + \epsilon_j \tag{2.1}$$

Die Grundlage der *funktionalen Datenanalyse* ist es, diese „beobachtete“ Funktion selbst als einzelne Beobachtung x zu betrachten. Besondere Betonung liegt dabei in der *Glattheitsanforderung* an die latente Funktion x. Ohne diese bestünde kein zusätzlicher Nutzen funktionaler gegenüber multivariater Betrachtung der Daten. Ist die Funktion x *glatt* in

[1] Für eine Einführung in die *funktionale Datenanalyse* siehe z.B. Ramsay and Silverman (2006) und Ramsay and Silverman (2002).

dem Sinn, dass sie eine oder mehrere Ableitungen besitzt, bedeutet dies, dass zwei aufeinanderfolgende Werte y_j und y_{j+1} zwangsläufig in einem gewissen Umfang miteinander verbunden sind und sich bei geringem Abstand nicht stark unterscheiden. Die diskreten Daten y_j, $j = 1, \ldots, T$ werden dabei üblicherweise dazu verwendet, die Funktion x und gleichzeitig eine bestimmte Anzahl ihrer Ableitungen zu schätzen.

Im Allgemeinen sieht man sich mit einer Stichprobe von funktionalen Daten konfrontiert und nicht nur mit einer einzelnen Beobachtung x. So besteht eine Beobachtung x_i, $i = 1, \ldots, n$ aus T_i Wertepaaren (t_{ij}, y_{ij}) mit $j = 1, \cdots, T_i$. Die Argumente t_{ij} können für jede Beobachtung gleich sein, aber genausogut von einer Beobachtung zur anderen variieren. Dies gilt auch für das Intervall $\mathcal{T}$, über dem die Daten gesammelt wurden, selbst.

KAPITEL 3

Anpassung funktionaler Daten

Bei den meisten praktischen Problemen liegen – wie schon erwähnt – von einer Funktion nur endlich viele diskret gemessene Werte $y_{i1}, \ldots, y_{in_i}$ vor und somit muss diese Funktion x_i mit Werten $x_i(t)$ als erstes interpoliert oder approximiert werden, um für jedes Argument t den zugehörigen Funktionswert berechnen zu können. Da die Messwerte häufig mit Fehlern behaftet sind und die zugrundeliegende Funktion durch den Messprozess verrauscht und dadurch nicht mehr erkennbar sein kann, ist es oft sinnvoll, bei der Schätzung der Funktion *Smoothing*-Techniken miteinzubeziehen und nicht ausschließlich zu interpolieren.
Beleuchtet man das Problem von einem theoretischen Standpunkt, hat man es bei funktionaler Datenanalyse mit unendlich-dimensionalen Daten wie Kurven zu tun. Um die Funktion in endlich-dimensionaler Weise darstellen zu können, ist eine Dimensionsreduktion notwendig. Dazu gibt es zwei gängige Möglichkeiten: Eine pragmatische Idee ist die Diskretisierung des Zeitintervalls, um Datenvektoren zu erhalten und die Anzahl der Messwerte auf einige wenige herunterzubrechen. Dagegen wird bei sogenannten Filterungsansätzen jede Kurve auf eine endlich-dimensionale Basis der Größe L projeziert und dadurch eine Dimensionsreduktion erreicht. Darunter fallen beispielsweise die *Splineexpansion* und die *Karhunen-Loève-Expansion*, die in den nachfolgenden Abschnitten näher erläutert werden.

3.1. Splineexpansion

Mit Hilfe einer Splineexpansion lässt sich nun die Schätzung einer Funktion auf ein endlich-dimensionales Problem reduzieren. Splines teilen sich mit der Anpassung von Polynomen den Vorteil, nur eine geringe Rechenintensität zum Speichern von Informationen von Kurven aufbringen zu müssen. Gleichzeitig sind aber Splines den Polynomen noch hinsichtlich der Flexibilität bei der Darstellungsmöglichkeit überlegen. Das sogenannte Basisfunktio-

nensystem einer Splineexpansion besteht aus einer Menge bekannter Funktionen ϕ_l, die unabhängig voneinander sind und die Eigenschaft besitzen, jede Funktion beliebig gut zu approximieren, indem eine ausreichend große Anzahl L dieser Funktionen als *Linearkombination* zusammengesetzt wird. Eine solche Funktionsanpassung lässt sich folgendermaßen ausdrücken:

$$x(t) = \sum_{l=1}^{L} \eta_l \phi_l(t), \tag{3.1}$$

mit η_l als zugehörige Basiskoeffizienten.
Bezeichnet man mit $\boldsymbol{\eta}$ den $L-$dimensionalen Koeffizientenvektor und mit $\boldsymbol{\phi}$ den Vektor bzw. die $n \times L$ – Matrix der Splinefunktionen $\phi_l(t_j)$ schreibt sich Gleichung 3.1 in Matrixnotation als:

$$x = \eta'\phi = \phi'\eta. \tag{3.2}$$

Bei der Wahl von $L = n$ können die Koeffizienten η_l so gewählt werden, dass $x(t_j) = y_j$ für alle j und somit eine exakte Darstellung der Kurve vorliegt. Im Gegensatz zu dieser Interpolation bestimmt sich bei einer Approximation der Grad, zu dem die Daten y_j geglättet werden, nach der Anzahl L an Basisfunktionen. Es wird deutlich, dass L selbst einen Parameter darstellt und nach der Beschaffenheit der Daten gewählt werden muss.

Die Schätzung der Splinekoeffizienten η_l kann mit Hilfe der Kleinste-Quadrate-Schätzung erfolgen. Die zu minimierende Zielfunktion *SSE* (*sum of squared errors*) lautet demnach:

$$SSE = \sum_{j=1}^{n} [y_j - \sum_{l}^{L} \eta_l \phi_l(t_j)]^2. \tag{3.3}$$

Als Schätzer für den Vektor der Basiskoeffizienten ergibt sich:

$$\hat{\eta} = (\Phi'\Phi)^{-1}\Phi y, \tag{3.4}$$

mit $\boldsymbol{y}$ als Vektor der tatsächlich beobachteten Werte.

Diese Schätzmethode ist in Situationen geeignet, in denen das Standard-Error-Modell angenommen wird, also die Residuen ϵ_j unabhängig und identisch verteilt sind, mit $\mathbb{E}(\epsilon_j) = 0$ und konstanter Varianz $Var(\epsilon_j) = \sigma^2$, $j = 1, \ldots, T$.

In der Praxis kommen bei der funktionalen Datenanalyse in den meisten Fällen Fourier-Basen für periodische und B-Spline-Basen oder natürliche Splines für nichtperiodische Daten zum Einsatz. Eine weitere Möglichkeit zum Anpassen von Funktionen stellen die sogenannten *Smoothing-Splines* dar. Diese genannten Möglichkeiten werden nachfolgend näher betrachtet.

3.1.1. Fourier-Basis-System

Bei der Fourier-Basis-Expansion für periodische Daten ergeben sich die Basisfunktionen als $\phi_0(t) = 1$, $\phi_{2r-1}(t) = \sin r\omega t$ und $\phi_{2r}(t) = \cos r\omega t$, wobei der Parameter ω die Periode $\frac{2\pi}{\omega}$ bestimmt. Die Darstellung als Basisfunktionenexpansion wie in (3.1) ist somit:

$$\hat{x}(t) = \eta_0 + \eta_1 \sin \omega t + \eta_2 \cos \omega t + \eta_3 \sin 2\omega t + \eta_4 \cos 2\omega t + \cdots \tag{3.5}$$

Ein Charakteristikum der Fourierbasen ist, dass sie, wenn die Werte t_j äquidistant auf dem Intervall $\mathcal{T}$ verteilt sind und die Periode gleich der Länge dieses Intervalls ist, *orthogonal* sind und somit die Kreuzproduktmatrix $\Phi'\Phi$ diagonal ist. Diese Eigenschaft spielt beim Clustern funktionaler Daten eine Rolle.

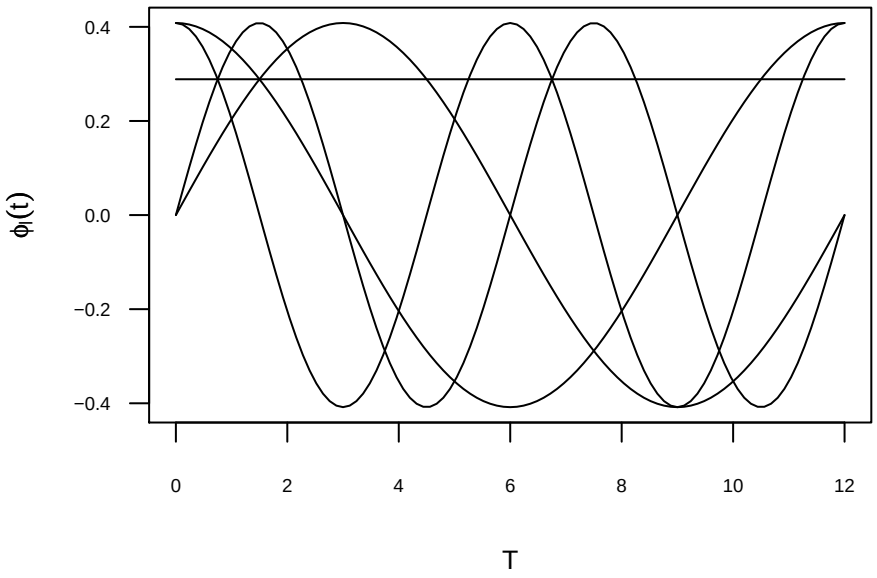


Abbildung 3.1.: Fourier-Basis mit fünf Basisfunktionen auf dem Intervall $\mathcal{T} = [0, 12]$

3.1.2. Polynomsplines

Etwas anders ist nun die allgemeine Struktur einer Splinefunktion bei der Anpassung nichtperiodischer Daten. Das Intervall $\mathcal{T}$, über dem die zu approximierende Funktion $x(t)$ definiert ist, wird hierbei in D Subintervalle geteilt. Diese Intervalle werden von den sogenannten *Knoten* τ_d, $d = 1, \ldots, D-1$, getrennt. Schließt man die Endpunkte als Knoten mit ein, ergeben sich die Knoten $\tau_0, \ldots, \tau_D$. Über jedem Subintervall ist ein Spline ein Polynom einer festgelegten Ordnung m bzw. vom Grad $m-1$. Die Knoten können dabei äquidistant gewählt, aber auch individuell gesetzt werden. Letzteres ist vor allem dann sinnvoll, wenn es Bereiche in der Funktion mit mehr Schwankungen und andere relativ geradlinige gibt. Darüberhinaus darf es keine Intervalle ohne Daten geben, da so der Funktionaverlauf aufgrund mangelnder Information nicht bestimmt werden kann.
Damit die einzelnen Polynome glatt ineinander übergehen, müssen die Funktionswerte an den Knoten gleich sein. Zusätzlich sind aufgrund der globalen Glattheitsanforderung bis zu $m-2$ Ableitungen an diesen Verbindungsstellen identisch. Zusammengefasst besteht eine Splinefunktion aus zwei Elementen, der Ordnung der Polynomstücke m und der Knotensequenz τ. Die Anzahl der Parameter, die $x(t)$ bestimmen, summiert sich aus Polynomordnung $m-1$ und der Anzahl innerer Knoten D zu $L = m + D - 1$.

Nach Fahrmeir et al. (2009) ist ein Polynomspline wie folgt definiert:

Definition 1 (Polynom-Spline) *Eine Funktion $f : [a, b] \to \mathbb{R}$ heißt Polynom-Spline vom Grad $m - 1 \geq 0$ zu den Knoten $a = \tau_0 < \cdots < \tau_d = b$, falls sie die folgenden Bedingungen erfüllt:*

1. *$f(t)$ ist $(m-2)$-mal stetig differenzierbar. Für $m = 2$ entspricht dies der Forderung, dass $f(t)$ stetig ist, für $m = 1$ werden keine Glattheitsanforderungen an $f(t)$ gestellt.*
2. *$f(t)$ ist auf den durch die Knoten gebildeten Intervallen $[\tau_j, \tau_{j+1})$ ein Polynom vom Grad $m - 1$.*

Deutlich wird, dass eine Funktionsanpassung durch Splineexpansion stark vom gewählten Splinegrad, sowie der Anzahl und Position der inneren Knoten abhängt. Eine in dem Sinn *richtige* Anpassung existiert nicht. Dennoch werden zumindest kubische Splines, also Polynomstücke vom Grad drei, standardmäßig verwendet, damit eine glatte, zweimal stetig differenzierbare Funktion entsteht. Ebenso ist bei vielen Anwendungen kein besonderer Funktionsverlauf in einzelnen Bereichen erkennbar und es werden äquidistante Knoten gewählt. Anders ist es bei der Knotenanzahl. Generell gilt, dass eine größere Anzahl an Knoten die Funktionsanpassung flexibler macht, während eine kleine Knotenanzahl in einer glatteren Anpassung mündet. Hier ist es oft notwendig theoretische Überlegungen zu einer adäquaten Wahl der Knotenanzahl anzustellen.

B-Spline-Basis Die äußerst bekannte B-Spline-Basis ist eine mögliche Darstellung solcher Polynomsplines. Die Funktion $x(t)$ lässt sich analog zu Gleichung (3.1) als B-Spline-Expansion darstellen:

$$\hat{x}(t) = \sum_{l=1}^{m+D-1} \eta_l B_l(t). \tag{3.6}$$

Im Gegensatz zu beispielsweise den Fourier-Basen sind B-Spline-Basen nur lokal definiert. Das bedeutet, dass eine B-Spline-Basisfunktion nur über $m+1$ benachbarten Intervallen existiert und an einer beliebigen Stelle des Intervalls $\mathcal{T}$ nur m Basisfunktionen positiv sind. Dadurch können auch Veränderungen in kleineren Bereichen deutlich gemacht werden. Insgesamt besteht die Basis aus m Polynomstücken vom Grad $m-1$, die an den Knoten $m-2$-mal stetig differenzierbar zusammengesetzt sind. Bei äquidistanten Knoten haben die mittleren Basisfunktionen dieselbe Form, während sie sich bei nicht-äquidistanten unterscheiden. Die $m-1$ Basisfunktionen am rechten und linken Rand von $\mathcal{T}$ distinguieren sich von den Übrigen. Am Rand von $\mathcal{T}$ steigt die Anzahl an aufeinanderfolgenden Intervallen, über denen die Basisfunktionen positiv sind, von eins zu m in Richtung des Zentrums an, besitzen aber immer noch den zweimal stetig differenzierbaren Übergang zu Null. Nach Außen hin verlieren die Basisfunktionen an Differenzierbarkeit, was die Tatsache unterstützt, dass der weitere Funktionsverlauf von $x(t)$ jenseits der Ränder nicht bekannt ist.

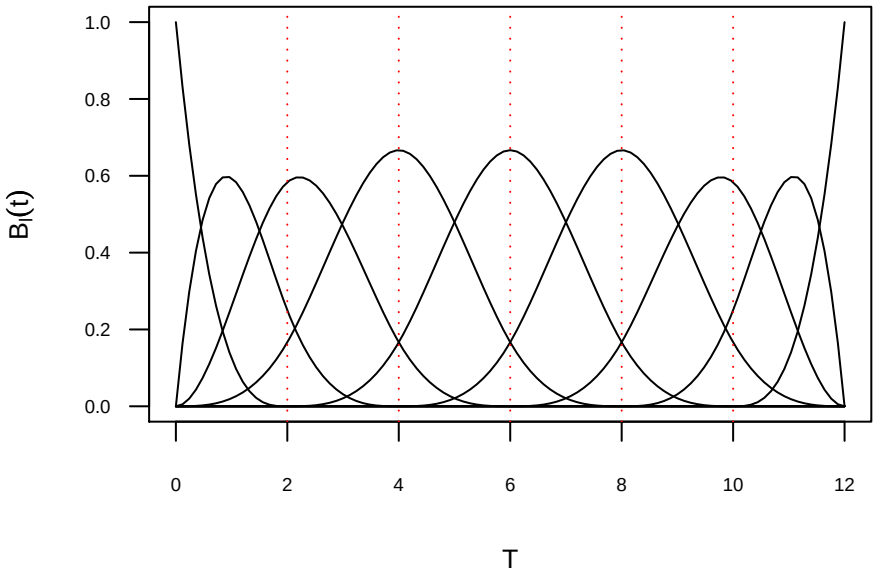


Abbildung 3.2.: B-Spline-Basis der Ordnung drei mit fünf inneren Knoten auf dem Intervall $\mathcal{T} = [0, 12]$

3.1.3. Smoothing Splines

Sind die zugrundeliegenden Daten nicht oder nicht annähernd gleichabständig, ist es sinnvoller, an jeden j-ten Datenpunkt einen Knoten zu setzen. Bei den Smoothing-Splines wird diese maximale Knotenmenge verwendet und dadurch Regionen mit hoher Datendichte begünstigt. Zudem erfolgen keine Annahmen über die nichtparametrische Funktion $x(t)$, außer dass sie zweimal stetig differenzierbar ist. Anders als vorher gibt es keine Beschränkung auf den durch Basisfunktionen gebildeten Funktionenraum mehr. Das SSE-Kriterium aus (3.3) erweitert sich um einen Strafterm, der zu raue Funktionsschätzungen, also eine zu starke Anpassung, vermeiden soll. Als Maß für die Krümmung einer Funktion wird ihre quadrierte zweite Ableitung verwendet. Die zu minimierende Zielfunktion bei den Smoothing Splines ist das *penalisierte Kleinste-Quadrate-Kriterium*:

$$PSSE = \sum_{j=1}^{n} [y_j - x(t_j)]^2 + \lambda \int [x''(t)]^2 dt \tag{3.7}$$

Aus der Gleichung wird deutlich, dass der erste Term die Nähe zu den Daten misst, während der zweite Term die Krümmung bestraft. Die Stärke der Bestrafung wird dabei durch den *Smoothingparameter* $\lambda \geq 0$ beeinflusst. Für $\lambda = 0$ kann $x(t)$ jede Funktion sein, die die Datenpunkte interpoliert, für $\lambda \to \infty$ ergibt sich ein Polynom vom Grad $m-1$ als Schätzung für $x(t)$. Gleichung (3.7) besitzt mit dem *natürlichen kubischen Spline* mit Knoten an den einzelnen Datenpunkten eine eindeutige, endlich-dimensionale Lösung. Nach Fahrmeir et al. (2009) ist ein *natürlicher kubischer Spline* wie folgt definiert:

Definition 2 (Natürlicher kubischer Spline) *Zu einer vorgegebenen Knotenmenge $a < \tau_0 < \cdots < \tau_d < b$ ist die Funktion f(t) genau dann ein natürlicher kubischer Spline, wenn gilt*

1. *$f(t)$ ist ein kubischer Polynom-Spline zur obigen Knotenmenge und*
2. *$f(t)$ erfüllt die Randbedingungen $f''(a) = f''(b) = 0$, d.h. $f(t)$ ist linear in den Intervallen $[a, \tau_0]$ und $[\tau_d, b]$.*

Ein natürlicher kubischer Spline ist demnach ein kubischer Polynomspline mit einer speziellen, der *natürlichen*, Randbedingung.

3.2. Karhunen-Loève-Expansion

Bei der Karhunen-Loève-Expansion[2] geht es darum, einen stochastischen Prozess als eine unendlich-dimensionale Linearkombination orthogonaler Funktionen zu beschreiben. Diese Darstellungsweise ähnelt der einer Fourierexpansion (vgl. Abschnitt 3.1.1) bei einer Funktionsanpassung auf einem beschränkten Intervall, nur dass hier die Koeffizienten Zufallsvariablen entsprechen. Ferner sind die Basisfunktionen der Expansion von der Funktion selbst abhängig. Genauer gesagt bestimmen sich die orthogonalen Basisfunktionen durch die Kovarianzfunktion des Prozesses und werden deshalb auch als „empirische“ orthogonale Funktionen bezeichnet. Formal stellt sich die Karhunen-Loève-Expansion wie folgt dar:

$$x(t) = \mu(t) + \sum_{l=1}^{\infty} \xi_l(x)\rho_l(t), \tag{3.8}$$

wobei
$\mu(t) = \mathbb{E}[x(t)]$ der marginale Mittelwert, ρ_l die Eigenfunktionen zur Kovarianz Γ mit $\Gamma(s,t) = cov\{x(s), x(t)\}$, und $\xi_l(x)$ zufällige, unkorrelierte Koeffizienten mit Mittelwert Null und Varianz λ_l sind.

Diese Repräsentation einer Funktion entspricht der einer funktionalen Hauptkomponentenanalyse[3]. Analog zum multivariaten Fall, in der es Ziel der Hauptkomponentenanalyse ist, die Eigenwerte und Eigenvektoren der Kovarianzmatrix aufzufinden, ist dies auch Aufgabe der funktionalen Betrachtung. Nur entsprechen die Eigenvektoren nun Eigenfunktionen und dienen als Basisfunktionen für die Expansion. Dementsprechend sind diese Basisfunktionen orthonormal, d.h. $\int \rho_j \rho_k = 0 \ \forall j < k$. Die Eigenfunktionen ρ_j zur Kovarianz Γ mit den zugehörigen Eigenwerten λ_j werden durch das Eigenwertproblem festgelegt, so dass jede Eigenwertfunktion, bzw. im Hauptkomponentenkontext, jede Principal-Component-Gewichtungs-Funktion $\rho_l(t)$ für einen bestimmten Eigenwert λ folgende Gleichung erfüllt:

$$\int \Gamma(s,t)\rho_l(t)dt = \lambda_l \rho_l(t) \tag{3.9}$$

Da der linke Teil der Gleichung eine sogenannte *Integraltransformation* Γ der Eigenfunktion ρ, definiert durch

$$\Gamma\rho = \int \Gamma(\cdot, t)\rho(t)dt, \tag{3.10}$$

ist, lässt sich das Eigenwertproblem wie im multivariaten Fall darstellen als

$$\Gamma\rho = \lambda\rho, \tag{3.11}$$

wobei ρ nun eine Funktion und keinen Vektor beschreibt.

[2] Vgl. Ramsay and Silverman (2006), Spanos (1991), Chiou and Li (2007), Chiou and Li (2008)
[3] Für eine Einführung in die funktionale Hauptkomponentenanalyse siehe Ramsay and Silverman (2006).

Die gesuchten Eigenwert-Eigenfunktions-Paare ergeben sich z.B. mit Hilfe diskreter Approximation durch Lösung der Gleichung $\int_0^T \hat{\Gamma}(s,t)\hat{\rho}_l(s)ds = \hat{\lambda}_l\hat{\rho}_l(t)$ unter der Bedingung $\int_0^T \hat{\rho}_j(t)\hat{\rho}_l(t)dt = 0$ für $j = t$ und 1 für $j \neq t$ (vgl. Chiou and Li (2007)). Andere Möglichkeiten finden sich bei Ramsay and Silverman 2006, Kapitel 8.4.

Die Koeffizienten $\xi_l(x)$ der Expansion sind unkorrelierte Zufallsvariable mit Mittelwert Null und Varianz λ_l, so dass

$$\xi_l(x) = \langle x - \mu, \rho_l \rangle \tag{3.12}$$

und damit eine Projektion der zentrierten Funktion $x - \mu$ in Richtung der l-ten Eigenfunktion ρ_l ist. Eine mögliche Approximation der funktionalen Principal-Component-Scores findet sich in Chiou and Li 2007, S.695, Anhang A.3.

KAPITEL 4

Clusterverfahren

Das Ziel einer Clusteranalyse ist es, ähnliche Objekte zu finden und diese zu Gruppen zusammenzuschließen. Die einzelnen Objekte sollten also so zusammengefasst werden, dass die Objekte im selben Cluster ähnlich zueinander und Objekte in unterschiedlichen Clustern unähnlich sind. Gemeinhin unterscheidet man zwischen zwei grundsätzlichen Vorgehensweisen, den heuristischen und den modellbasierten Ansätzen.

4.1. Heuristische Ansätze

Beim Clustern funktionaler Daten kommen i.d.R. unter den heuristischen Ansätzen nichthierarchische partitionierende Verfahren wie der bekannte k-means bzw. k-medoids Algorithmus zum Einsatz. Bei den *partitionierenden* Clusterverfahren wird – ausgehend von einer festgesetzten Clusteranzahl – eine vorgegebene Startlösung, also eine anfängliche Klasseneinteilung der Daten, durch schrittweises Verschieben von Objekten zwischen den Clustern so verändert, dass ein Heterogenitätskriterium $H(\mathscr{C})$ für eine Partition $\mathscr{C}$ optimiert wird. Die beste Partition unter zwingender Vorgabe der Clusteranzahl wird bestimmt durch $H(\mathscr{C}_{opt}) = \min_{\mathscr{C}} H(\mathscr{C})$. Vorteil nicht-hierarchischer Verfahren ist die Austauschbarkeit der Objekte zwischen den Clustern, die bei hierarchischen Verfahren nicht gegeben ist. Kritisch bei den partitionierenden Clustermethoden ist dagegen die Beeinflussung der Ergebnisse durch die subjektiv festgelegte Startposition und die Zielfunktion. So können immer nur lokale Optima erreicht werden, keine globalen.

k-means Die gängigste Methode unter den partitionierenden Clusterverfahren ist der *k-means-Algorithmus*. Dieser Algorithmus hat die Aufgabe, n Objekte mit je T Messungen in

eine fest vorgegebene Anzahl K Gruppen $(C_1, C_2, \ldots, C_K)$ zuzuordnen. Als C_k bezeichnet sich die Menge der n_k Objekte im Cluster k. Die Partition wird dadurch gebildet, dass die Distanz zwischen dem Datenvektor $\mathbf{x}_i$ und dem Centroidvektor des jeweiligen Clusters $c(\mathbf{x}_i)$ minimiert wird:

$$d(\mathbf{X}_i, C_k) = \frac{1}{n} \sum_{i=1}^{n} d(\mathbf{x}_i, c(\mathbf{x}_i)) \rightarrow \min_{C_k} \tag{4.1}$$

Dabei berechnet sich der Centroid in Cluster C_k als Durchschnitt der jeweiligen Variablenwerte T über alle Objekte, die sich im Cluster befinden. So ergibt sich der Centroidwert für die j-te Variable im k-ten Cluster als

$$\bar{x}_j^{(k)} = \frac{1}{n_k} \sum_{i \in C_k} x_{ij} \tag{4.2}$$

Der vollständige Centroidvektor $\bar{\mathbf{x}}^{(k)} = (\bar{x}_1^{(k)}, \bar{x}_2^{(k)}, \ldots, \bar{x}_T^{(k)})$ für das Cluster C_k setzt sich aus den einzelnen Durchschnittswerten der Variablen zusammen. Das iterative Vorgehen beim k-means-Algorithmus[4] läuft wie folgt ab:

Algorithmus 1 (k-means-Algorithmus)

1. *Starte mit K zufälligen T-dimensionalen Centroidvektoren $\mathbf{c}^{(k)} = (c_1^{(k)}, c_2^{(k)}, \ldots, c_T^{(k)})$.*
2. *Berechne die Distanz $d(i,k)$ zwischen dem i-ten Objekt und dem k-ten Cluster mit der quadrierten euklidischen Distanz:*

$$d(i,k) = \sum_{j=1}^{T} (x_{ij} - c_j^{(k)})^2 \tag{4.3}$$

   Vergleiche die Objekte mit jedem (neuen) Centroid bzgl. ihrer Distanz $d(i,k)$ zueinander und ordne sie ihrem nächstgelegenen Cluster zu.
3. *Ermittle nach dieser (anfänglichen) Zuordnung der Objekte die neuen Centroide für jedes Cluster gemäß (4.2).*
4. *Wiederhole die Schritte 2 bis 3 solange, bis kein Objekt mehr zwischen den Clustern wechselt.*

Der *k-means Algorithmus* startet also mit K Clustermittelwerten, die i.d.R. zufällig aus den Datenpunkten gezogen werden. Danach werden alle einzelnen Datenpunkte demjenigen Cluster zugeordnet, das bzgl. der quadrierten euklidischen Distanz am nächsten liegt. Die Clustermittelwerte werden nach der vollständigen Zuordnung aller Datenpunkte als Durchschnittswert aller sich im jeweiligen Cluster befindenden Objekte neu berechnet und

[4] in Anlehnung an Steinley 2006

der Algorithmus solange iteriert, bis keine Objekte mehr zwischen den Clustern wechseln. Insgesamt wird bei der Optimierung der Partition versucht, die Fehlerquadratsumme (SSE) zu minimieren:

$$SSE = \sum_{j=1}^{T} \sum_{k=1}^{K} \sum_{i \in C_k} (x_{ij} - \bar{x}_j^{(k)})^2 \tag{4.4}$$

Innerhalb des k-means-Algorithmus ist es auch möglich, andere Distanzen als die euklidische Distanz zu verwenden. Die allgemeine Vorgehensweise ohne konkrete Festlegung des Distanzmaßes kann als *verallgemeinerter k-means-Algorithmus* für ein k-Centroid-Clusterproblem bezeichnet werden.[5]

k-medians Eine Alternative zum klassischen k-means Algorithmus ist der k-medians Algorithmus (vgl. Steinley (2006) und Leisch (2006)). Dieser ist weniger empfindlich gegenüber Ausreißern. Der Unterschied dieses speziellen k-Centroid-Algorithmus zum k-means besteht also im Distanzmaß. Während k-means die quadrierte eunklidische Distanz 4.3 verwendet, bedient sich der k-medians Algorithmus der Manhattan-Distanz:

$$d(i,k) = \sum_{j=1}^{T} \left| x_{ij} - c_j^{(k)} \right|. \tag{4.5}$$

Als Verlustfunktion ergibt sich die Fehlerquadratsumme als

$$SSE = \sum_{j=1}^{T} \sum_{k=1}^{K} \sum_{i \in C_k} \left| x_{ij} - med_j^{(k)} \right|. \tag{4.6}$$

Die Centroide entsprechen demnach dem Median der Beobachtungen innerhalb des jeweiligen Clusters k.

[5] siehe Leisch 2006

4.2. Modellbasierte Verfahren

Die zweite große Klasse beim Clustern sind die modellbasierten Verfahren.[6] Hierbei wird angenommen, dass die Beobachtungen $x_1, \ldots, x_n$ Realisationen des Zufallsvektors $\mathbf{x}$ sind und $\mathbf{x}$ in jeder Klasse eine andere Verteilung besitzt. Die Daten stammen also aus einer sogenannten *Mischverteilung* mit K Komponenten, wobei die Klassenzugehörigkeiten z_i nicht bekannt sind. Grundsätzlich gibt es zwei mögliche Vorgehensweisen die Clusterzugehörigkeiten z_i zu behandeln. Dies mündet in zwei unterschiedliche Ansätze, der *classification likelihood* und der *mixture likelihood*.

classification likelihood Bei der *classification likelihood* werden die z_i als Parameter gesehen und das Modell durch die Maximierung der folgenden Likelihood angepasst:

$$L_C(\theta_1, \cdots, \theta_K; \mathbf{z}_1, \cdots, \mathbf{z}_n | \mathbf{x}_1, \cdots, \mathbf{x}_n) = \prod_{i=1}^{n} f_{\mathbf{z}_i}(\mathbf{x}_i | \theta_{\mathbf{z}_i}) \tag{4.7}$$

mit $f_{z_i}(x_i|\theta_{z_i})$ als Dichte von Cluster z_i mit den zugehörigen Paramtern im Cluster θ_{z_i}.

mixture likelihood Betrachtet man die Clusterzugehörigkeit z_i dagegen als fehlenden Wert und nimmt an, dass z_i multinomial verteilt ist mit Parametern $(\pi_1, \ldots, \pi_K)$ mit π_k als Wahrscheinlichkeit, dass eine Beobachtung zum k-ten Cluster gehört, dann lautet die zu maximierende sogenannte *mixture likelihood*:

$$L_M(\theta_1, \cdots, \theta_K; \pi_1, \cdots, \pi_K | \mathbf{x}_1, \cdots, \mathbf{x}_n) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k f_k(\mathbf{x}_i | \theta_k). \tag{4.8}$$

Dabei stellt $f_k(x_i|\theta_k)$ die Dichte und θ_k den unbekannten Parametervektor der $k-$ten Mischungskomponeten, sowie π_k mit $\pi_k \geq 0$ und $\sum_{k=1}^{K} = 1\pi_k$ den unbekannten Mischungsanteil dar.

Die Schätzung der Parameter erfolgt mit Hilfe des EM-Algorithmus (Dempster et al. (1977)) für Mischverteilungen (vgl. Fraley and Raftery (2002) und Fahrmeir and Heumann (0809)).

[6] siehe Banfield and Raftery 1993, Fraley and Raftery 2002 und James and Sugar 2003

EM-Algorithmus für Mischverteilungen

Notation:

- y_i beobachtete (unvollständige) Daten
- z_i unbeobachtete Daten bzw. latente Variablen
- $x_i = (y_i, z_i)$ vollständige Daten

Die Idee des EM-Algorithmus zur ML-Schätzung bei unvollständigen Daten ist es, nicht die Likelihood der beobachteten Daten, sondern die der vollständigen Daten zu maximieren. Dazu alterniert der Algorithmus zwischen zwei Schritten, dem *Expectation-Step* (**E**-Step) und dem *Maximization-Step* (**M**-Step):

Algorithmus 2 (EM-Algorithmus)
Wähle Startwerte der Parameter $\theta^{(0)}$.

***E**-Step:*
Berechne den bedingten Erwartungswert der Log-Likelihood der vollständigen Daten, gegeben die beobachteten Daten und die aktuellen Parameterschätzer:

$$Q(\theta) = Q(\theta|\theta^{(0)}) = \mathbb{E}\left[l(\theta; y, z|y, \theta^{(0)})\right]$$

***M**-Step:*
Schätze die Parameter $\theta^{(1)}$*, die diese erwartete Log-Likelihood des **E**-Step maximieren:*

$$Q(\theta^{(1)}) = arg\max_{\theta} Q(\theta)$$

*Iteriere **E**-/**M**-Steps bis zur Konvergenz.*

Bei Mischverteilungen sind – wie bereits erwähnt – die unbekannten, latenten Variablen die Klassenzugehörigkeiten $z_i = (z_{i1}, \ldots, z_{iK})$ mit

$$z_{ik} = \begin{cases} 1, & \text{falls } x_i \text{ zur Klasse k gehört} \\ 0, & \text{sonst.} \end{cases}$$

Die Log-Likelihood der beobachteten Daten y ist

$$l(\theta; y) = \sum_{i=1}^{n} log\left[\sum_{k=1}^{K} \pi_k f_k(y_i|\theta_k)\right] \tag{4.9}$$

mit π_k – wie zuvor definiert – als unbekannte Mischungsanteile.

Die Log-Likelihood der vollständigen Daten $x = (y, z)$ enthält die unbekannte Indikatorvariable z_i und ergibt sich als

$$
\begin{aligned}
l(\theta; y, z) &= \sum_{i=1}^{n} log f(x_i, z_i|\theta) && (4.10)\\
&= \sum_{i=1}^{n} \log[f(x_i|z_i;\theta) \cdot f(z_i)] && (4.11)\\
&= \sum_{i=1}^{n} [\log f_{z_i}(x_i|\theta_{z_i}) + \log \pi_{z_i}] && (4.12)
\end{aligned}
$$

Der Algorithmus läuft wie folgt ab:

Algorithmus 3 (EM-Algorithmus für Mischverteilungen)
Wähle geeignete Startwerte $\theta_1, \ldots, \theta_K$ und a-priori Wahrscheinlichkeiten $\pi_1, \ldots, \pi_K$.

***E**-Step:*

Berechne für $i = 1, \ldots, n$ die a-posteriori Wahrscheinlichkeiten

$$
\begin{aligned}
\hat{p}_{ik} &= \mathbb{E}\left[I(z_i = k)|y_i, \theta\right] && (4.13)\\
&= \frac{\hat{\pi}_k f_k(y_i|\hat{\theta}_k)}{\sum_{m=1}^{K} \hat{\pi}_m f_m(y_i|\hat{\theta}_m)} && (4.14)
\end{aligned}
$$

für $k = 1, \ldots, K$ und daraus die a-priori Klassenwahrscheinlichkeit

$$\hat{\pi}_k = \frac{1}{n} \sum_{i=1}^{n} \hat{\pi}_{ik}. \qquad (4.15)$$

***M**-Step:*

Berechne $\hat{\theta}_k|z$ für $k = 1, \ldots, K$.

Somit wird im **E**-Step des EM-Algorithmus die Indikatorvariable durch ihren Erwartungswert $\mathbb{E}(z_i)$ ersetzt. Der **M**-Step dient zur Maximierung der Log-Likelihood der vollständigen Daten bzgl. π_{z_i} und θ_{z_i}, wobei z_{ik} fest auf den im **E**-Step berechneten Wert gesetzt wird. Analog werden wiederum die **E**- und **M**-Schritte solange iteriert, bis der Algorithmus konvergiert.

KAPITEL 5

Methoden zum Clustern funktionaler Daten

Auch bei longitudinal erhobenen funktionalen Daten wird oft nach repräsentativen Kurvenverläufen gefragt. Deshalb ist eine der Aufgaben der funktionalen Datenanalyse das Clustern solcher Kurven. Bisherige Ansätze lassen sich als Kombinationen von Dimensionsreduktion zur endlich-dimensionalen Darstellbarkeit der Funktionen und dem Clusterverfahren, ob modellbasiert oder heuristisch, darstellen (siehe Tabelle 5.1). Im folgenden Abschnitt soll ein grober Überblick über diese bisherigen Verfahren gegeben werden.

	heuristische Verfahren (k-means)	modellbasierte Verfahren
Orthonormale Basis	Serban and Wasserman (2005) Shimizu and Mizuta (2008) Tarpey and Kinateder (2003) Tarpey et al. (2003) Tarpey (2007a)	
B-Splines	Abraham et al. (2003) García-Escudero and Gordaliza (2005) Jank and Shmueli (2006) (penalisiert, k-medians) Tarpey (2007a) Hitchcock et al. (2007) (k-medoids mit PAM)	Luan and Li (2003)
natürliche Splines		James and Sugar (2003)
Smoothing Splines		Ma et al. (2006)
Karhunen-Loève-Expansion	Chiou and Li (2007) Chiou and Li (2008)	

Tabelle 5.1.: Übersicht der Clusterverfahren für funktionale Daten

5.1. Modellbasierte Verfahren

Zu den modellbasierten Verfahren gehören die Ansätze von James and Sugar (2003), Luan and Li (2003) und Ma et al. (2006). Alle drei Ansätze basieren auf einem sogenannten *Mixed-Effects-Model*[7] innerhalb eines Clusters. Grundlegende Idee dabei ist, dass sich eine funktionale Beobachtung aus einer typischen, oft der Mittelwertskurve des jeweiligen Clusters und einer zufälligen kurvenspezifischen Abweichung zuzüglich eines Messfehlers zusammensetzt. So wird nicht für jeden Beobachtungsvektor separat eine Splinekurve angepasst, sondern dieser Beobachtungsvektor und alle Beobachtungen im selben Cluster dazu verwendet, einen populationsspezifischen Verlauf zu modellieren. Durch das Poolen von Daten aus einem Cluster werden zuverlässigere Schätzer der Kurvenverläufe erwartet, als bei Verfahren, die nur die einzelnen Daten verwenden.

Ein übliches Mixed-Effects-Model für eine Beobachtung zum Zeitpunkt t_{ij}, gegeben die beobachtete Kurve i ist in Cluster k, lässt sich schreiben als:

$$y_i = X_i\beta + Z_i\gamma_i + \epsilon_i \tag{5.1}$$

wobei β den Vektor der fixen Effekte und γ_i den Vektor der zufälligen Effekte darstellt. Das bekannte lineare Modell wird damit um einen weiteren Term $Z_i\gamma_i$ ergänzt, der dazu dient, die erwähnte Abweichung jeder Kurve von ihrem populationsbezogenen mittleren Kurvenverlauf zu berücksichtigen. Normalerweise wird dabei angenommen, dass die zufälligen Effekte i.i.d. $\gamma_i \sim N(0, \Gamma)$ und die Fehler i.i.d. und unabhängig von γ_i $\epsilon_i \sim N(0, \sigma^2 I)$ verteilt sind.

Die einzelnen Verfahren unterscheiden sich in der Modellierung des festen Effekts, aber vor allem in der des zufälligen Effekts. So wird beim *Functional Clustering Model* (FCM) von James and Sugar (2003) sowohl für die Modellierung der populationsspezifischen Kurve eines Clusters als auch für die Zufallsabweichung eine Splineexpansion $y_i(t) = S_i\eta_i + \epsilon_i$ mit $\eta_i = \mu_{z_i} + \gamma_i$ gewählt:

$$y_i(t) = S_i(t)\mu_{z_i} + S_i(t)\gamma_i + \epsilon_i \tag{5.2}$$

mit S_i als natürliche Splinebasis, $\gamma_i \sim N(0, \Gamma)$ und $\epsilon_i \sim N(0, \sigma^2 I)$.
Bei Luan and Li (2003) wird die typische Kurve im k-ten Cluster sowie die zufällige Abweichung mit Hilfe von B-Spline-Basen dargestellt:

$$y_i(t_{ij}) = \sum_{l=1}^{p} \beta_l^{(k)} \bar{B}_l(t_{ij}) + \sum_{l=1}^{q} \gamma_{il} B_l(t_{ij}) + \epsilon_{ij} \tag{5.3}$$

mit denselben Verteilungsannahmen für γ_i und ϵ_i.

[7] Für eine Beschreibung von *Mixed-Effects-Models* siehe Laird and Ware (1982).

Die beiden Ansätze von James and Sugar (2003) und Luan and Li (2003) clustern anschließend die Kurven mit Hilfe des EM-Algorithmus für entweder die classification likelihood (vgl. (4.7)) oder Mischverteilungsmodelle (vgl. (4.8)). Ma et al. (2006) verwenden bei ihrem *Smoothing Spline Clustering* (SSC) Smoothing Splines zur Kurvenanpassung der populationsspezifischen Kurve in einem bestimmten Cluster und unterstellen eine zusätzliche kurvenspezifische Verschiebung, die über alle Zeitpunkte gleich ist. Die Parameter und die Clusterzugehörigkeit schätzen sie unter Annahme einer mixture likelihood (vgl.4.8) mit einer Abwandlung des EM-Algorithmus, dem *Rejection-controlled EM*.

5.2. Heuristische Ansätze

Weit verbreitet beim Clustern funktionaler Daten ist die Approximation der Kurven mittels Splineexpansion mit anschließendem Clustern der geschätzten Splinekoeffizienten der individuellen Kurven mit klassischen heuristischen Verfahren wie k-means (vgl. Abschnitt 4.1). Die heuristischen Ansätze lassen sich nach unterschiedlichen Clustermethoden – k-means und k-medians – sowie nach ihrer Kurvenanpassung mittels Splineexpansion (vgl. Abschnitt 3.1) mit Verwendung orthonormaler Basen, B-Splines, natürlichen Splines oder Smoothing Splines gegenüber einer Karhunen-Loève-Expansion (vgl. Abschnitt 3.2) abgrenzen.

Orthonormale Basisexpansion und k-means Clustern Eine Kombinationsmöglichkeit von Splineexpansion und Clustern stellt eine orthonormale Basisexpansion der Funktionen, verbunden mit dem Clustern der geschätzten Splinekoeffizienten mit dem k-means-Algorithmus dar. Diese Vorgehensweise findet sich u.a. bei Tarpey and Kinateder (2003), Tarpey et al. (2003), Serban and Wasserman (2005), Tarpey (2007a) und Shimizu and Mizuta (2008). Indem zur Approximation der Funktionen eine orthonormale Basisexpansion gewählt wird, können funktionale Daten als Koordinaten im L-dimensionalen Raum dargestellt werden und die euklidische Distanz zwischen zwei Koordinaten in diesem Raum entspricht der Distanz zwischen den zwei entsprechenden funktionalen Daten (vgl. Tarpey and Kinateder (2003) und Tarpey (2007a)). Als orthonormale Basen werden häufig orthonormale Polynome oder Fourierbasen (vgl. Abschnitt 3.1.1) verwendet und die Splinekoeffizienten in üblicher Weise mit der KQ-Methode geschätzt. Diese Splinekoeffizienten stellen dann die zu clusternden Objekte x_{ij} im k-means-Algorithmus aus Abschnitt 4.1 dar. Beim Ansatz von Serban and Wasserman (2005) wird der Problematik des Einschlusses von zu flachen Kurvenverläufen im Clusteralgorithmus Rechnung getragen. Nach der Approximation der Kurven durch eine Fourierexpansion werden hier mit einem nichtparametrischen Test konstante Kurven aus den Daten entfernt und anschließend erst die Splinekoeffizienten geclustert.

Karhunen-Loève-Expansion und k-means-Clustern Chiou and Li (2007) und Chiou and Li (2008) kritisieren, dass bei den Splineexpansionen immer die gleichen Basisfunktionen für alle Daten angesetzt werden müssen. Dagegen verwenden ihre Ansätze jeweils die Eigenbasen der Kurven zur Prozessexpansion, die sogenannte Karhunen-Loève-Expansion (vgl. Abschnitt 3.2). Die *k-centres FC*-Methode von Chiou and Li (2007) basiert darauf, dass die beobachteten Kurven aus einer Mischung stochastischer Prozesse stammen und jeder dieser Unterprozesse jeweils eine (trunkierte) Karhunen-Loève-Expansion besitzt. Somit lässt sich jede beobachtete Kurve x mit Hilfe der trunkierten Karhunen-Loève-Expansion durch die L ersten der eigentlich unendlich vielen Eigenfunktionen darstellen, womit Gleichung (3.8) zu

$$x^{(k)}(t) = \mu^{(k)}(t) + \sum_{l=1}^{L_k} \xi_l^{(k)}(x)\rho_l^{(k)}(t) \tag{5.4}$$

wird. Falls nun Kurve x tatsächlich zu Cluster k gehört, entspricht $x^{(k)}(t)$ der Karhunen-Lóève-Expansion von $x(t)$, ist dies nicht der Fall unterscheiden sich die beiden Funktionen. Das Clustern der Kurven erfolgt ähnlich zum k-means Algorithmus. Im ersten Schritt werden die marginalen FPC-Scores (functional principal component scores) aller Kurven $\hat{\xi} = (\hat{\xi}_{i1}, \ldots, \hat{\xi}_{iL})$ mit dem k-means-Algorithmus geclustert und man erhält für eine vorgegebene Clusteranzahl K die Klassenzugehörigkeiten der einzelnen Kurven. Im zweiten Schritt werden nun die Objekte iterativ reklassifiziert. Als optimale Clusterzuordnung für eine Kurve dient als Kriterium die L^2-Distanz zwischen der Kurve und der Clusterzentrumskurve $\tilde{x}^{(k)}$, dargestellt als Karhunen-Loève-Expansion wie in Gleichung (5.4):

$$k^*(x) = arg \min_{k \in \{1,\ldots,K\}} \left\| x - \tilde{x}^{(k)} \right\| \tag{5.5}$$

Die neuen Clusterzentren berechnen sich anschließend als Expansion aller Kurven im jeweiligen Cluster.

Ein weiteres Verfahren von Chiou und Li (vgl. Chiou and Li (2008)) fokussiert sich auf die Kurvenform selbst. So sollen Funktionen, die sich nur durch eine vertikale Verschiebung und durch einen multiplikativen Effekt bzgl. der Skalierung unterscheiden, demselben Cluster zugeordnet werden. Dafür entwickelten sie ein *functional multiplicative random-effects model* in Verbindung mit einer Karhunen-Loève Expansion. Die Idee ist ähnlich dem vorhergehenden Ansatz. Falls x_i tatsächlich zum Cluster k gehört, ist die Approximation im Cluster k gleich der trunkierten stochastischen Expansion mit M_k Komponenten, verbunden mit einem multiplikativen Skaleneffekt. Die Ähnlichkeit wird in diesem Fall durch die funktionale Korrelation zwischen x_i und $x_{i(c)}^{M_k}$ gemessen.

B-Spline-Basisexpansion und k-Centroid-Clustern Die populärste Methode beim Clustern funktionaler Daten ist wohl die Kurveninterpolation mit B-Splines (vgl. Abschnitt 3.1.2) und anschließendem Clustern der geschätzten Splinekoeffizienten mit einem k-medoid-Algorithmus, oft dem klassischen k-means-Verfahren. Diese Vorgehensweise findet sich mehr oder weniger in ihrer Ursprungsform belassen in Abraham et al. (2003), García-Escudero and Gordaliza (2005), Jank and Shmueli (2006), Tarpey (2007a) und Hitchcock et al. (2007) (vgl. auch Tabelle 5.1). Der Anpassungsgrad an die Kurve wird dabei vom Splinegrad und von der festgelegten Knotenmenge bestimmt. Oft werden, wie bereits in Abschnitt 3.1.2 erwähnt, in der Praxis kubische B-Splines gewählt, da diese meist eine ausreichende Beschreibung eines stetigen und glatten Prozesses erlauben. Die Funktion lässt sich somit wieder durch einige wenige Parameter, die Splinekoeffizienten, charakterisieren, auf deren Basis die Kurven den entsprechenden Clustern mit Hilfe des k-means-Algorithmus zugeordnet werden. Die so entstandenen Clusterzentren bestehen ihrerseits wieder aus Splinekoeffizienten, die multipliziert mit der B-Spline-Basis die Verläufe der Clusterzentren ergeben.

Nicht selten gibt es in vorliegenden Daten Ausreißer, die – gerade bei k-means, das für seine Ausreißerempfindlichkeit bekannt ist – sehr starken Einfluss auf die Clusterergebnisse nehmen können, so auch bei funktionalen Daten. Deshalb kann es nützlich sein, solche auffälligen Daten aufzudecken und aus dem Clusterprozess auszuschließen. García-Escudero and Gordaliza (2005) stellen dazu das *Robust Curve Clustering* vor, das mit dem sogenannten *trimmed k-means*[8] arbeitet. Beim *trimmed k-means* werden die Clusterzentren nur aus $[n(1-\alpha)]$ Datenpunkten berechnet, wobei $\alpha \in [0,1]$ der sogenannte Trimmungsparameter ist, der angibt, wieviel Prozent der Daten weggeschnitten werden. Die Trimmung selbst ergibt sich aus den jeweiligen Daten und deren Kontaminationslevel. Der typische *trimmed k-means Algorithmus* läuft für ein vorgegebenes α wie folgt ab (in Anlehnung an García-Escudero et al. 2003):

Algorithmus 4 (trimmed k-means-Algorithmus)

1. *Starte mit K zufälligen T-dimensionalen Centroidvektoren $\mathbf{c}^{(k)} = (c_1^{(k)}, c_2^{(k)}, \ldots, c_T^{(k)})$.*
2. *Berechne die Distanz jeder Beobachtung zu ihrem nächstgelegenen Zentrum gemäß:*

$$d(i,k) = \sum_{j=1}^{T} (x_{ij} - c_j^{(k)})^2 \tag{5.6}$$

und behalte die Menge H mit insgesamt $[n(1-\alpha)]$ Beobachtungen mit den niedrigsten Distanzen $d(i,k)$.

Teile H in $H = \{H_1, \ldots, H_K\}$, wobei die Punkte in H_k diejenigen sind, die näher an c_k liegen als zu einem anderen Zentrum.

[8] Für eine genauere Einführung zu *trimmed k-means* siehe García-Escudero and Gordaliza (2005) und García-Escudero et al. (2003).

3. Wiederhole Schritt 2 einige Male.
Nach dieser Iteration, berechne die endgültige Zielfunktion

$$\frac{1}{[n(1-\alpha)]} \sum_{j=1}^{T} \sum_{i \in H_k} \sum_{l=1}^{K} (x_{ij} - c_j^{(k)})^2 \tag{5.7}$$

Ebenfalls ein abweichendes Distanzmaß bei ansonsten gleicher Vorgehensweise verwenden Hitchcock et al. (2007). Die Distanz basiert hier auf paarweisen Unähnlichkeiten und lässt sich bei n Messungen über einen Zeitraum $\mathcal{T} = [t_1, \ldots, t_T]$ schreiben als:

$$d(i,k) = \frac{t_T - t_1}{T-1} \left\{ \frac{\hat{x}_i(t_1) - \hat{x}_k(t_1)}{2} + \sum_{j=1}^{T-1} [\hat{x}_i(t_j) - \hat{x}_k(t_j)] + \frac{\hat{x}_i(t_T) - \hat{x}_k(t_T)}{2} \right\}, \tag{5.8}$$

wobei $\hat{x}$ die geglättete Kurve darstellt. Jank and Shmueli (2006) weichen hingegen sowohl im Distanzmaß als auch in der Splineanpassung vom Standardverfahren ab. So verwenden sie anstelle des k-means Algorithmus den in Abschnitt 4.1 ebenfalls beschriebenen k-medians Algorithmus, zur Kurvenanpassung verwenden sie *P-Splines* mit selbst festgelegten, nicht-äquidistanten Knoten und Smoothingparameter λ. Die Kurvenanpassung erfolgt hier im Grunde vergleichbar mit der bei den Smoothing Splines aus Abschnitt 3.1.3. Ziel ist es, das penalisierte KQ-Kriterium ähnlich Gleichung (3.7):

$$PSSE = \sum_{j=1}^{T} [y_j - \sum_{l=1}^{L} \eta_l B_l(t_j)]^2 + \lambda \sum_{l=d+1}^{L} (\Delta^d \eta_l)^2 \tag{5.9}$$

zu minimieren, wobei mit Δ^d Differenzen d-ter Ordnung bezeichnet werden.[9]

[9] Für eine genauere Beschreibung von P-Splines siehe Fahrmeir et al. 2009.

KAPITEL 6

Erweiterungen zum Clustern funktionaler Daten mit dem k-medoids-Algorithmus

6.1. Poissondistanz

Der in Abschnitt 4.1 vorgestellte *verallgemeinerte k-means* Algorithmus lässt unterschiedliche Distanzen zum Berechnen der Clustercentroide zu. Die Wahl der Distanz sollte dabei nicht zuletzt an den Daten orientiert sein. Der klassische k-means Algorithmus verwendet, wie in der Box auf Seite 14 dargestellt, die euklidische Distanz zum Berechnen des Abstands zu den Centroiden. Handelt es sich bei den zu clusternden Daten um Zähldaten, so ist ein symmetrisches Maß nicht unbedingt sinnvoll. Um der Schiefe der Verteilung Rechnung zu tragen, ist es denkbar, ein neues Distanzmaß im Clusteralgorithmus zu verwenden. So erscheint es naheliegend, die Likelihood der per Annahme poissonverteilten Zähldaten miteinzubeziehen.

Da die Ähnlichkeit der Beobachtungen mit zunehmender Likelihood steigt, sollte als Distanzmaß, als Maß der Unähnlichkeit, die negative Log-Likelihood der beobachteten Daten passend sein. Nimmt man die Beobachtungen über die Zeitpunkte als unabhängig an, ergibt die negative Log-Likelihood l aufsummiert über alle Zeitpunkte ein entsprechendes Distanzmaß. Somit berechnet sich die Distanz der beobachteten Daten x_{ij} zum jeweiligen Centroidvektor $c_j^{(k)}$ im Cluster k durch:

$$d(i,k) = \sum_{j=1}^{T} \left[-l(x_{ij}, c_j^{(k)}) \right] \tag{6.1}$$

$$= \sum_{j=1}^{T} -log \left[f(x_{ij}|c_j^{(k)}) \right] \tag{6.2}$$

und unter der Annahme von poissonverteilten Daten:

$$d(i,k) = \sum_{j=1}^{T} \left[-log \left(\frac{c_j^{(k)^{x_{ij}}}}{x_{ij}!} e^{-c_j^{(k)}} \right) \right] \tag{6.3}$$

Die neuen Centroide berechnen sich bei poissonverteilten Daten analog zu 4.2. Der *k-means Algorithmus* läuft ansonsten gleich ab.[10]

6.2. Splineclustern

Eine andere Möglichkeit als – wie in Abschnitt 5.2 beschrieben – die Splinekoeffizienten von Kurven zu clustern, ist es, die Splineexpansion in den Clusteralgorithmus selbst mitaufzunehmen. Dabei wird anstelle des Centroidvektors ein Spline durch die Centroidpunkte gelegt und danach an den Messzeitpunkten ausgewertet. Ein typischer Splineclusteralgorithmus mit k-means-Verfahren würde wie folgt aussehen:

Algorithmus 5 (Splineclustern)

1. *Starte mit K zufälligen T-dimensionalen Centroidvektoren $c^{(k)} = (c_1^{(k)}, c_2^{(k)}, \ldots, c_T^{(k)})$.*

2. *Lege einen Spline durch den Centroidvektor:*

$$\mathbf{c}^{(k)} = \eta^{(k)} S \tag{6.4}$$

Schätze die Splinekoeffizienten mittels KQ-Schätzung:

$$\hat{\eta}^{(k)} = (S'S)^{-1} S' c^{(k)} \tag{6.5}$$

und berechne den Splinewert an den einzelnen Messzeitpunkten:

$$c_j^{(k),neu} = \sum_{l=1}^{L} \hat{\eta}_l^{(k)} S_l(t) \tag{6.6}$$

Berechne die Distanz $d(i,k)$ zwischen dem i-ten Objekt und dem k-ten Clustercentroid. Vergleiche die Objekte mit jedem (neuen) Centroid bzgl. ihrer Distanz $d(i,k)$ zueinander und ordne jede Beobachtung dem nächstgelegenen Cluster zu.

3. *Ermittle nach dieser (anfänglichen) Zuordnung der Objekte die neuen Centroidvektoren für jedes Cluster gemäß (4.2):*

$$\bar{x}_j^{(k)} = \frac{1}{n_k} \sum_{i \in C_k} x_{ij}.$$

4. *Wiederhole die Schritte 2 bis 3 solange, bis kein Objekt mehr zwischen den Clustern wechselt.*

[10] Die Poissondistanz ist in einer Erweiterung des `flexclust` package als Funktion `distPoisson` hinterlegt, ©Friedrich Leisch.

Als Distanzen $d(i,k)$ sind beispielsweise die euklidische Distanz, aber auch die in Abschnitt 6.1 beschriebene Poissondistanz denkbar.

Beispiel An einem Beispiel soll kurz der Unterschied des $k-$means-Clusterns der Beobachtungsvektoren, der Splinekoeffizienten bzw. Splinewerte und dem Splineclustern aufgezeigt werden. Es wird angenommen, dass die rote Kurve in Abbildung 6.1 einen typischen Verlauf über die Zeit t darstellt und die schwarzen Linien verrauschte Beobachtungsvektoren sind, die diesem Muster folgen. Der Centroidvektor, der sich innerhalb eines Schrittes beim $k-$means-Algorithmus gemäß Formel (4.2) ergeben würde, ist als durchgehende blaue Linie in der Abbildung eingezeichnet. Das Splineclustern geht nun noch einen Schritt weiter und legt durch diesen Mittelwert einen Spline, um dem glatten Verlauf des zugrundeliegenden Musters Rechnung zu tragen.

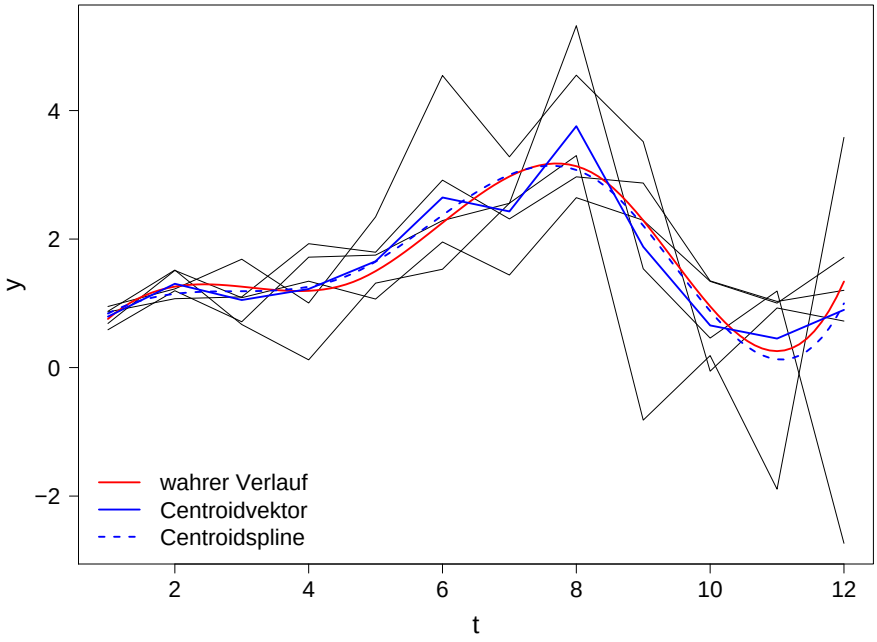


Abbildung 6.1.: Beobachtungen eines Verlaufsmusters mit Centroidvektor und Centroidspline

Jetzt wird exemplarisch ein Beobachtungsvektor herausgegriffen. Dieser ist in Abbildung 6.2 als schwarze durchgehende Linie geplottet.

Die gängige Methode funktionale Daten zu clustern, beinhaltet als ersten Schritt die Splineexpansion der Beobachtungsvektoren. Diese ist für den speziellen Datenvektor als gestrichelte schwarze Linie dargestellt. Weiterhin beruht der partitionierende $k-$means-

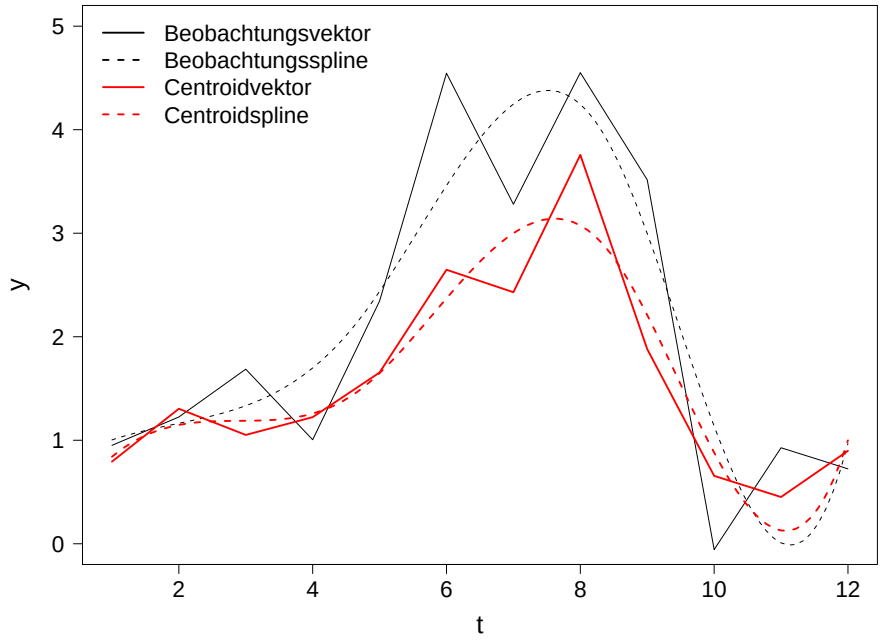


Abbildung 6.2.: Ausgewählte Beobachtung eines Verlaufsmusters mit Centroidvektor und Centroidspline I

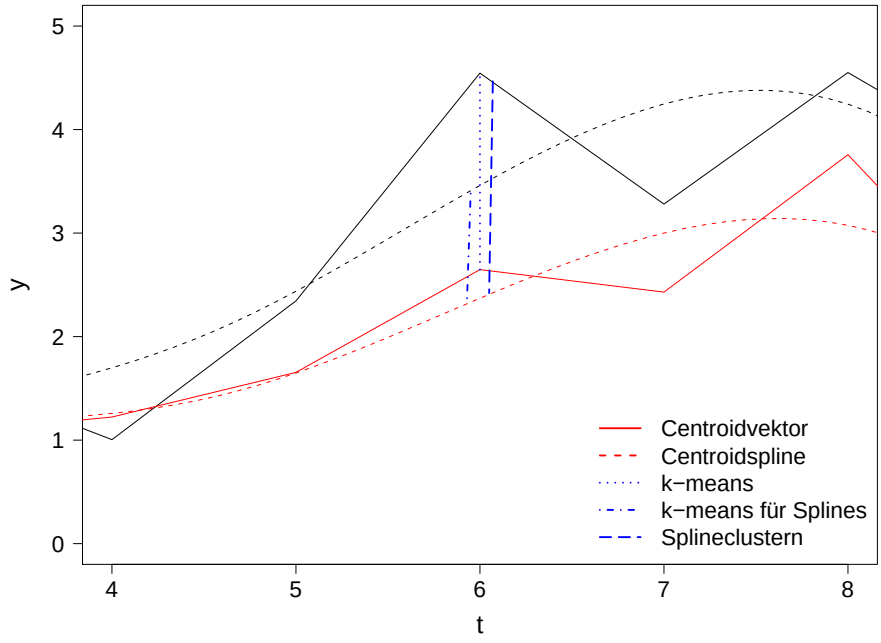


Abbildung 6.3.: Ausgewählte Beobachtung eines Verlaufsmusters mit Centroidvektor und Centroidspline II

Algorithmus auf der Distanz der einzelnen Beobachtung zu den Centroiden. Hier im Beispiel ist der Centroid der Beobachtungsvektoren als rote Linie und die Splineexpansion dieses Mittelwertvektors – die für das Splineclustern benötigt wird – als rot-gestrichelte Linie eingezeichnet.
Abbildung 6.3 zeigt, welche Verläufe jeweils zur Berechnung der Distanz der Beobachtung zum Centroid miteinander in Beziehung gesetzt werden. Beim Clustern der Rohdaten werden die einzelnen Beobachtungsvektoren mit dem jeweiligen Mittelwert der Beobachtungsvektoren innerhalb eines Clusters verglichen (blau-gepunktete Linie). Das Clustern der Splinekoeffizienten oder auch der Splinewerte selbst beruht auf der Distanz der Splines zum Mittelwert dieser Splines. Das neue Verfahren dagegen verwendet die Distanz der einzelnen Beobachtungsvektoren zu dem mit Hilfe von Splines expandierten Mittelwertsvektor. Dadurch ergeben sich jeweils unterschiedliche Distanzen und – damit einhergehend – abweichende Clusterzuordnungen und Centroidverläufe der Beobachtungen.

KAPITEL 7

Funktionales Clustern von Transaktionsverläufen

In diesem Kapitel sollen nun verschiedene, partitionierende Clustermethoden, basierend auf dem k-means Algorithmus aus Abschnitt 4.1 anhand realer und simulierter Daten durchgeführt werden. Ziel ist es, herauszufinden, ob und wie sich die betrachteten Clusterverfahren für die vorliegenden Daten, nämlich Transaktionsverläufe, unterscheiden. Wie verhält sich das Standardverfahren der funktionalen Clusteranalyse – k-means Clustern von Basiskoeffizienten einer Splineexpansion – bei dieser Art von Daten? Bringt die neue Poissondistanz aus Abschnitt 6.1 einen Vorteil gegenüber der herkömmlich verwendeten euklidischen Distanz und wie wichtig ist es, die Funktionalität der Daten überhaupt zu berücksichtigen?
Zunächst werden in 7.1 die betrachteten Daten und die zum Auffinden von typischen Verläufen verwendeten Clustermöglichkeiten vorgestellt. Der darauffolgende Abschnitt 7.2 beschäftigt sich mit der Analyse von realen Transaktionsdaten und dabei auch mit der Frage nach der Ähnlichkeit der Verfahren. Für die Analyse der besten Partition sind, da bei echten Daten die wahre Clusterzugehörigkeit unbekannt ist, simulierte Daten notwendig. Hier ist es möglich, zu entscheiden, inwieweit es dem jeweiligen Clusteralgorithmus gelungen ist, die wahre zugrundeliegende Struktur der Daten zu erkennen. Deshalb erfolgt in Abschnitt 7.3 eine Untersuchung anhand selbst erzeugter Daten. Abschließend werden in 7.4 die Ergebnisse für reale und simulierte Daten zusammengefasst und geklärt, welche Verfahren für Transaktionsverläufe hier am sinnvollsten erscheinen.

7.1. Daten und Vorgehen

Als Ausgangspunkt für die Analyse lagen Informationen von 4424 Haushalten über deren Einkäufe von 65 verschiedenen Produkten bzw. Produktgruppen innerhalb eines Kalenderjahres vor. Von jedem Haushalt (im folgenden auch mit HH abgekürzt) wurden Kalendertag und Anzahl der jeweils bei diesem Einkauf erworbenen Produkte aufgezeichnet. Aus den 65 Produktkategorien erfolgte für die weitere Untersuchung eine Auswahl von vier Typen:

- Produkt 61: Mineralwasser
- Produkt 44: Alkoholfreie Getränke ohne Kohlensäure (Fruchthaltige)
- Produkt 63: Eiscreme (Haushaltspackungen)
- Produkt 19: Zahnpasta

Bei *Mineralwasser*, *alkoholfreien Getränken* und *Eiscreme* sollten, nach eigener Überlegung, mehr Schwankungen bei der Anzahl der Einkäufe über die Sommer- und Wintermonate auftreten, als bei *Zahnpasta*, bei deren Transaktionsverlauf eher ein konstanteres Verhalten über die Zeit vermutet wurde.
Um äquidistante Messungen der Einkaufsmenge zu erhalten, wurden in einem ersten Schritt alle Einkäufe der jeweiligen Produkte getrennt für jeden Haushalt sowohl auf Kalenderwochenebene (im folgenden auch mit KW abgekürzt), als auch auf Monatsebene aggregiert. Eine Betrachtung von Transaktionen auf 2-Wochen-Ebene wurde aufgrund ähnlicher Muster beim Kaufverhalten wie auf Kalenderwochenebene nicht weiter verfolgt. Abbildung 7.1 zeigt beispielsweise die Verteilung der Anzahl der Einkäufe von Produkt 61 – *Mineralwasser* aller Haushalte aufgeteilt nach den 12 aufgezeichneten Monaten.

Im vorliegenden Fall stellen die Einkäufe eines Haushaltes zu einem Zeitpunkt oder aggregiert über einen bestimmten Zeitraum Zähldaten dar. Die Verteilung der Einkaufsmenge aller Haushalte sieht über die einzelnen Monate betrachtet jeweils für ein bestimmtes Produkt sehr ähnlich aus. Klar erkennbar ist eine Dominanz sehr niedriger Einkaufsmengen und die damit einhergehende rechtsschiefe Verteilung der Transaktionszahl. Oft werden die Produkte gar nicht oder nur einmal pro Woche bzw. Monat besorgt. Die durchschnittliche Anzahl an gekauften Produkten pro Kalenderwoche liegt bei Produkt 19 – *Zahnpasta* ungefähr bei 0.2, bei den Getränken (Produkt 44 – *Alkoholfreie Getränke (Fruchthaltige)* und Produkt 61 – *Mineralwasser*) bei jeweils 0.4 und bei der *Eiscreme* (Produkt 63) bei 0.1. Auf Monatsebene führen ebenfalls die *Getränke* mit einem durchschnittlichen Einkaufsvolumen von 1.6 bzw. 1.7, die *Zahnpasta* wird ca. einmal pro Monat und die *Eiscreme* im Schnitt 0.5 mal eingekauft. Das aus Abbildung 7.1 deutlich werdende Einkaufsverhalten und die damit verbundene Schiefe der Verteilung ist bei allen vier Produkten zu erkennen.[11] Somit

[11] Die Histogramme für alle vier Produkte auf Kalenderwochen-, 2-Wochen- und Monatsebene finden sich in Anhang F.

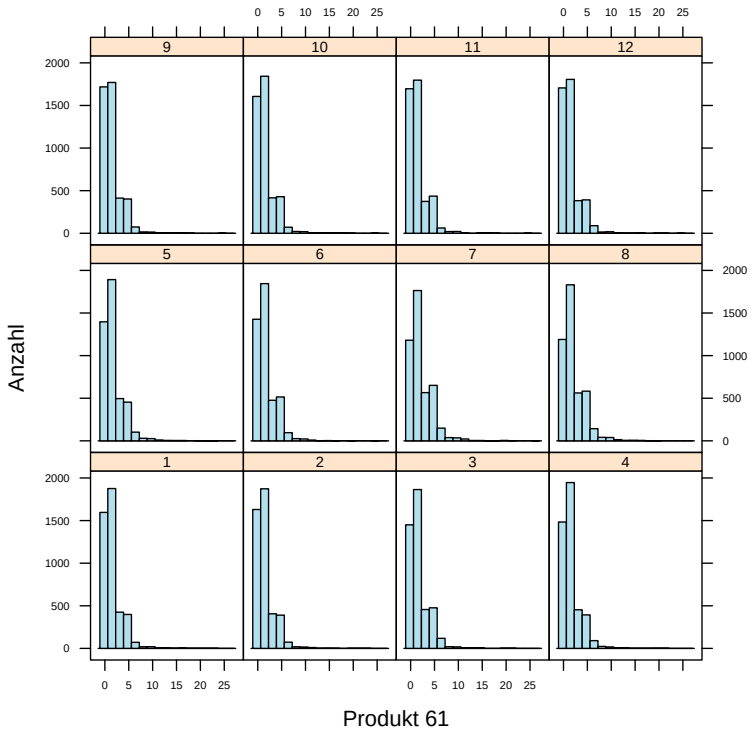


Abbildung 7.1.: Histogramm der Einkäufe des Produktes 61 nach Monat

ist es fragwürdig gängige k-means-Algorithmen, die auf symmetrischen Distanzmaßen beruhen, zum Clustern dieser Daten zu verwenden. Ein Fokus der weiteren Betrachtung ist demzufolge das in Abschnitt 6.1 vorgestellte Distanzmaß, das die theoretische Verteilung der Daten berücksichtigen kann.
Neben der mangelnden Symmetrie existiert eine weitere Besonderheit von Transaktionsdaten. So lässt sich bei der Betrachtung von Einkaufsverhalten ein glatter Verlauf der getätigten Transaktionen eines Haushaltes, also ein funktionaler Verlauf, unterstellen. Grundgedanke dabei ist, dass die Anzahl eines gekauften Produkts von vorherliegenden und nachfolgend geplanten Käufen beeinflusst ist und einem gewissen, glatt darstellbaren Einkaufsmuster folgt. Die unterschiedliche Behandlung der Daten im Hinblick auf die Einbeziehung der funktionalen Form ist demnach beim Clustern von Transaktionsdaten ebenfalls von großem Interesse. Entweder bleibt die Funktionalität dabei gänzlich unberücksichtigt oder sie wird in Form von Splines zur Darstellung der Kurven (vgl. Abschnitt 3.1) innerhalb oder im Vorfeld des Clusterns bedacht.

Mit den in Abschnitt 6 besprochenen Erweiterungen entstehen sechs weiter verfolgte Möglichkeiten die zugrundeliegenden Daten unter Verwendung des $k-$means-Algorithmus, kombiniert mit den verschiedenen Distanzmaßen (euklidisches vs. Poisson) sowie Vorgehensweisen (klassisches k-means-Verfahren vs. Splineclustern), zu partitionieren:

Möglichkeiten zum Clustern:

1. Clustern der Rohdaten mit k-means und euklidischer Distanz
2. Clustern der Rohdaten mit k-means und Poisson-Distanz
3. Clustern der Koeffizienten der Splineexpansion mit k-means und euklidischer Distanz
4. Clustern der mittels Splineexpansion vorhergesagten Werte an den Messzeitpunkten mit k-means und euklidischer Distanz
5. Clustern der Rohdaten mit Splineclustern und euklidischer Distanz
6. Clustern der Rohdaten mit Splineclustern und Poisson-Distanz

Bei Methode 1 und 2 bleibt die funktionale Form der Daten unberücksichtigt. Geclustert werden die Beobachtungen, die mehrdimensional in Form von äquidistanten bzw. als äquidistant aggregierte Messungen vorliegen. Im betrachteten Fall wären die aufzuteilenden Beobachtungen jeweils die Vektoren, die die Anzahl an Einkäufen des betrachteten Produktes eines Haushaltes innerhalb jeder Kalenderwoche (52 Wochen) bzw. Monat (12 Monate) beinhalten. Die dadurch entstehenden 4424 52- bzw. 12-dimensionalen Beobachtungsvektoren werden mit Hilfe des k-means Algorithmus partitioniert. Bei Methode 1 wird dabei auf die standardmäßig verwendete euklidische Distanz zurückgegriffen, was im klassischen k-means-Algorithmus mündet, während Methode 2 mit der neu entwickelten Poisson-Distanz (vgl. Abschnitt 6.1) arbeitet.[12] Methode 3 stellt das Standardverfahren beim Clustern funktionaler Daten dar. Hierbei werden analog zu Abschnitt 5.2 zu *B-Spline-Basisexpansion und k-means-Clustern* die Beobachtungsvektoren zunächst geglättet und anschließend die geschätzten Basiskoeffizienten mit dem k-means Algorithmus zu Gruppen zusammengeschlossen. Alternativ wäre auch denkbar, nicht die Koeffizienten zu clustern, sondern die vorhergesagten Werte der Funktionsanpassung an den Messzeitpunkten selbst (Methode 4). So erfolgt im Grunde bei Verfahren 4 gegenüber Methode 1 eine Glättung der Rohdaten. Das Splineclustern als weitere Vorgehensweise, die funktionale Form von Daten zu berücksichtigen, wird bei Methode 5 und 6 angewandt. Der Unterschied zwischen Clustermethode 5 und 6 ist wiederum das Distanzmaß, 5 verwendet klassisch die euklidische Distanz und 6 die Poisson-Distanz zur Messung des Abstandes zwischen den Beobachtungen und den Clustercentroiden.

Sowohl für die Anpassung der Kurven mit Splines, als auch für das Splineclustern wurden kubische B-Splines (vgl. Abschnitt 3.1.2 zu B-Splines) gewählt. Nach inhaltlichen Überlegungen über abzubildende Einkaufsmuster, erfolgte eine Festlegung der Anzahl an Frei-

[12] Die Berechnungen erfolgten mit einer bisher unveröffentlichten Erweiterung des Pakets `flexclust`, ©Freidrich Leisch, in `R` unter Verwendung der Funktion `stepFlexclust`.

heitsgraden auf $df = 6$, wobei sich bei kubischen B-Spline-Basen zwei innere Knoten auf dem betrachteten Intervall ergeben. Mit zwei inneren Knoten und Polynomstücken vom Grad 3 lassen sich ausreichend Extrema darstellen, um saisonale Effekte im Einkaufsverhalten abbilden zu können.

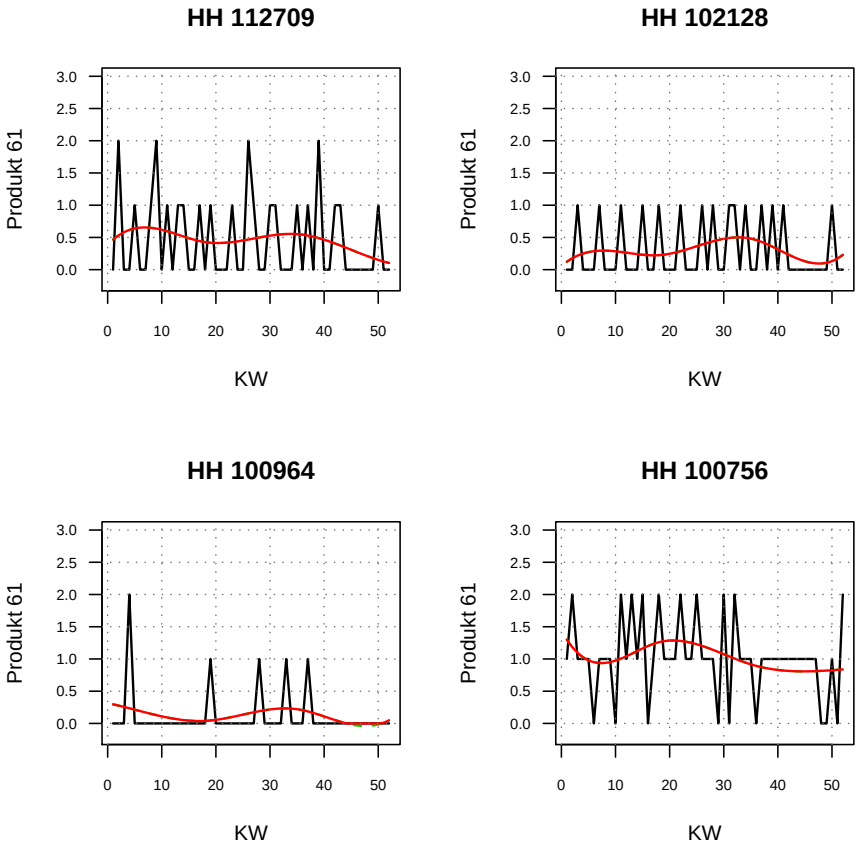


Abbildung 7.2.: Anzahl gekauftes Produkt 61 über die Zeit mit angepasstem B-Spline der Ordnung 3 und 2 inneren Knoten, KW-Ebene

Bei der Splineexpansion im Vorfeld zur Clusteranalyse wurde, als erster Schritt, durch jeden Beobachtungsvektor eines Haushaltes ein Spline gelegt. Abbildung 7.2 zeigt die Ausgangskurve der Einkäufe von Produkt 61 – *Mineralwasser* vier einzelner Haushalte mit zugehöriger Splineexpansion in rot eingezeichnet. Mit der Wahl von nur zwei inneren Knoten erfolgt eine eher starke Glättung, die das Erkennen von groben Strukturen zulässt. So zeigt beispielsweise Graphik 7.2, dass die Haushalte 112709, 102128 und 100964 ähnliche Muster in ihrem Kaufverhalten für *Mineralwasser* (Produkt 61) aufweisen. Es gibt sowohl im Frühjahr einen mehr oder weniger stark ausgeprägten Anstieg, der um die Kalenderwochen 15 – 22, also April bis Mai, zurückgeht und im Sommer bis zum Herbst hin wieder stärker ansteigt. Dagegen wird bei Haushalt 100756 das ganze Frühjahr sowie den Sommer

über stärker eingekauft und in den Wintermonaten schwächer.

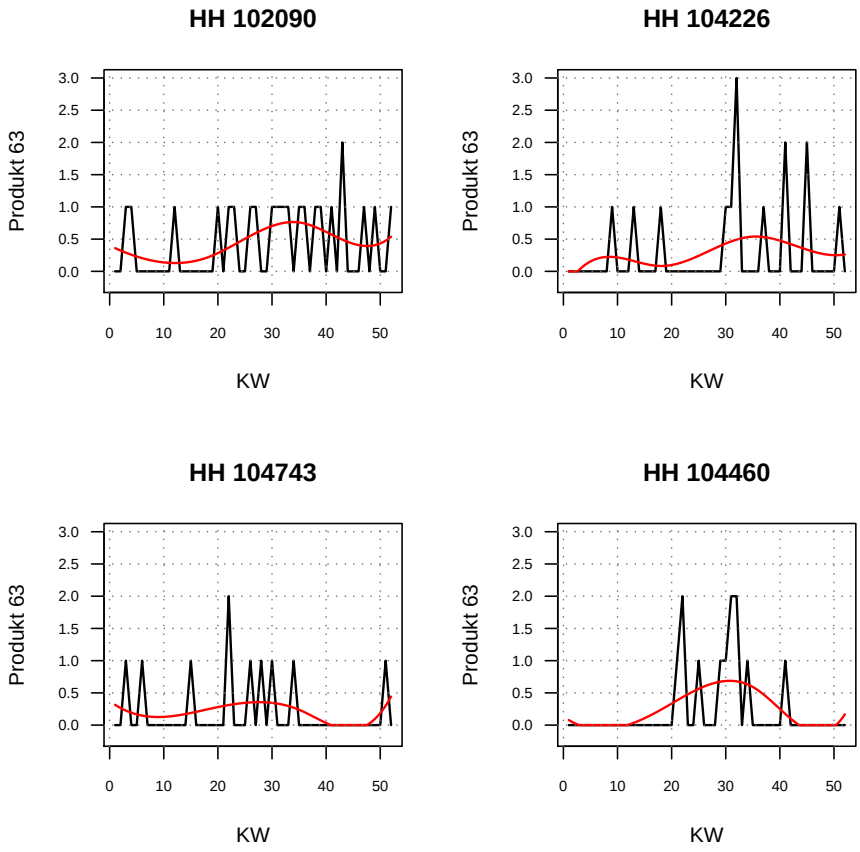


Abbildung 7.3.: Anzahl gekauftes Produkt 63 über die Zeit mit angepasstem B-Spline der Ordnung 3 und 2 inneren Knoten, KW-Ebene

Bei Produkt 63 – *Eiscreme* auf KW-Ebene lassen sich in Abbildung 7.3 bei den Haushalten 102090 und 104743 ähnliche Verläufe erkennen. Beide weisen im Frühjahr einen niedrigen Transaktionsverlauf auf, der, nach einem stetigen Anstieg ab Ende des Frühjahrs, um KW 30 herum sein Maximum erreicht und danach wieder abflacht. Anders ist die Struktur bei Haushalt 104226. Dieser zeigt zwei Höhepunkte beim Einkaufsmuster auf, einen im Frühjahr, Mitte März, und einen im Hochsommer, Ende Juli bis Anfang August. Haushalt 104460 dagegen kauft ab dem späten Frühjahr bis in den frühen Herbst hinein *Eiscreme*, mit der maximalen Einkaufsmenge im Hochsommer.

Aus den beiden Abbildungen 7.2 und 7.3 soll deutlich werden, dass die Anpassung eines geeigneten Splines mit ausreichend inneren Knoten von der Fragestellung abhängt. In unserem Fall, in dem die einzelnen Käufe eine untergeordnete Rolle spielen und nur ein allgemeiner Trend im Vordergrund steht, ist eine so geringe Knotenanzahl ausreichend. Mit der Beschränkung auf die hier gewählte Knotenanzahl von zwei können beispielsweise etwaige Gruppen saisonaler Einkaufsmuster aufgefunden werden. Differenziertere Aussagen

auf beispielsweise wöchentlichen Änderungen sind nicht zu treffen.

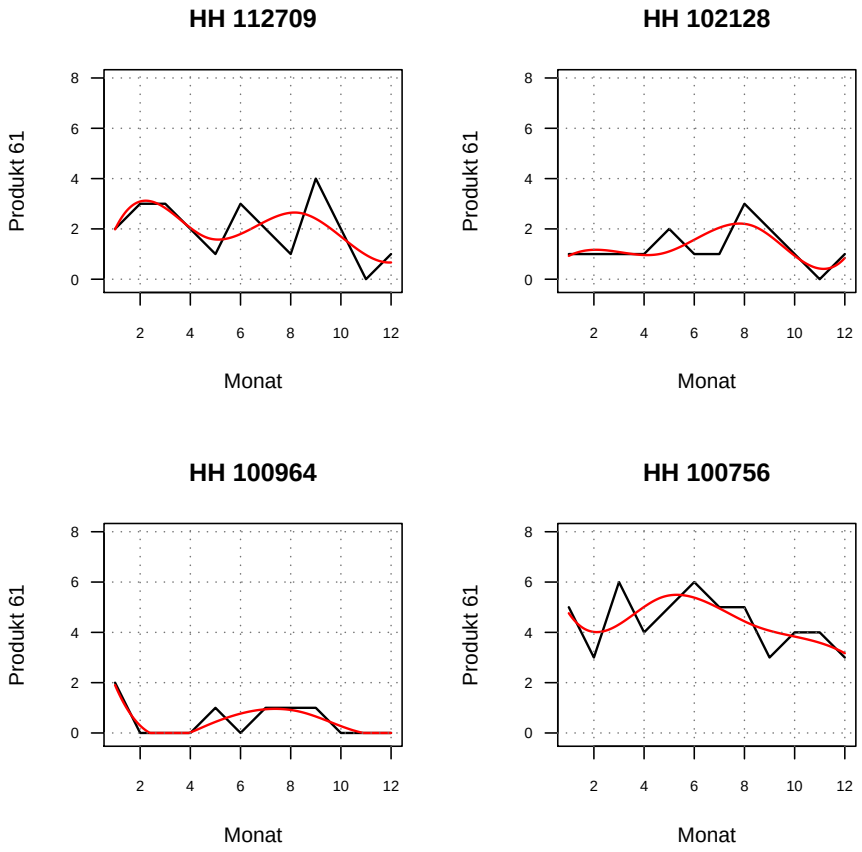


Abbildung 7.4.: Anzahl gekauftes Produkt 61 über die Zeit mit angepasstem B-Spline der Ordnung 3 und 2 inneren Knoten, Monatsebene

Um stärkere Muster zu erkennen, ist es auch denkbar, neben einer wöchentlichen Zusammenfassung von Einkäufen eine noch höhere Aggregationsstufe zu betrachten. Abbildung 7.4 zeigt eine Zusammenfassung der Einkäufe von Kalenderwochenebene auf Monatsebene für Produkt 61 – *Mineralwasser* derselben Haushalte wie zuvor. Hier ist klar erkennbar, dass sich mögliche saisonale Muster gegenüber der feineren Betrachtung noch verstärken. Bei den Haushalten 102128 und 100964 kommt nun der Einkaufsanstieg in den Sommermonaten heraus, während sich bei Haushalt 112709 sowohl ein Frühjahrs-, als auch ein Sommerhoch abzeichnet.

Nach Festlegung von Splinegrad und Knotenanzahl wurden für alle Produkte die vorgestellten sechs Clustermöglichkeiten, sowohl auf Kalenderwochen- als auch auf Monatsebene, durchgeführt. Für partitionierende Clusterverfahren ist es zudem – wie aus Abschnitt 4.1 bekannt – notwendig, die Anzahl der aufzufindenden Klassen im Vorfeld zum Clustern festzulegen. Die Anzahl der Cluster wurde dabei auf $k = 4$ gesetzt. Die Wahl erfolgte

auf Basis eines *Scree-Plots*, der als graphisches Hilfsmittel zur Festlegung einer geeigneten Gruppenanzahl bei partitionierenden Clusteralgorithmen dient.

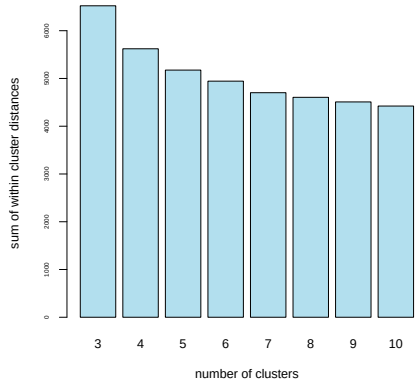


Abbildung 7.5.: Scree-Plot für $k = 3$ bis $k = 10$ Cluster bei Clusterverfahren 4, Produkt 61, KW-Ebene

Beim *Scree-Plot* wird eine variierende Klassenanzahl k gegen das zu minimierende Zielkriterium geplottet. Das Zielkriterium ist monoton fallend in der Klassenanzahl. Als graphisch bestimmte, optimale Clusteranzahl gilt der Wert k, der gegenüber der nächstkleineren Klassenanzahl den größten Sprung in der Abbildung macht und damit die größte Reduktion des Zielkriteriums aufweist. In den meisten Fällen zeigte diese Graphik, wie in Abbildung 7.5 exemplarisch für Produkt 61 – *Mineralwasser* bei Clustermethode 4 abgebildet, den größten Sprung von drei auf vier Cluster an. Mit steigender Klassenanzahl ergaben sich keine größeren Sprünge der Verlustfunktion mehr.[13] Somit werden nachfolgend in Abschnitt 7.2 kurz die Ergebnisse der Clusteralgorithmen für die realen Daten auf Kalenderwochen- und Monatsebene bei einer festgelegten Clusteranzahl von vier dargestellt.

[13] Die *Scree-Plots* für die einzelnen Clusteralgorithmen finden sich in Teilen in Anhang F.

7.2. Reale Transaktionsdaten

7.2.1. Verläufe der Clusterzentren

7.2.1.1. Produkt 63 – Eiscreme (Haushaltspackungen)

Clusterverfahren 1 Auf Kalenderwochenebene ergaben sich beispielsweise für Produkt 63 – *Eiscreme* für das Clustern der Rohdaten mit k-means und der euklidischen Distanz die in Abbildung 7.6 dargestellten Clusterzentren. Die durch den Algorithmus aufgedeckten Centroide sind dabei Vektoren, deren Elemente in der Abbildung aufgrund des zeitlichen Verlaufs miteinander verbunden wurden. Somit entsteht rein optisch der Eindruck eines zeitlichen Verlaufs. Erkennbar ist, dass keinerlei Glättung erfolgt und die Clusterzentren sehr rau sind. Die Abbildung zeigt drei sehr ähnliche Transaktionsmuster, die sich lediglich in der Höhe der Einkäufe, nicht aber in deren zeitlichem Verlauf unterscheiden.

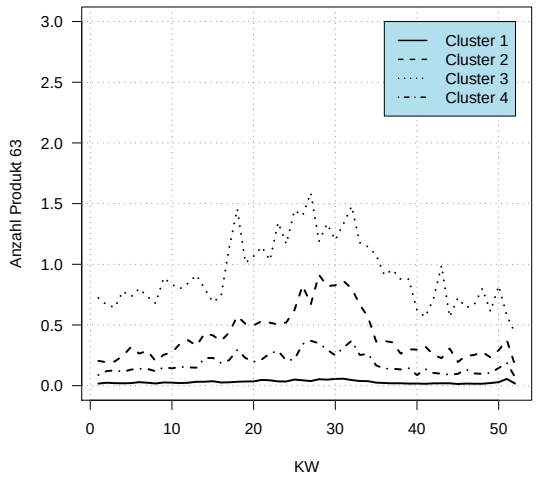


Abbildung 7.6.: Verlauf der Clusterzentren, Clustern der Rohdaten mit k-means und euklidischer Distanz, Produkt 63, KW-Ebene

Das vierte Zentrum in Abbildung 7.6 liegt um die 0.2, das zweite schwankt um 0.5 und das dritte um einen Einkauf pro Woche. Alle diese drei Zentren weisen eine leicht ansteigende Konsummenge bis ins späte Frühjahr auf, mit dem Maximum im Hochsommer. Danach fällt die Einkaufsmenge in den Herbst- und Wintermonaten auf ein Basisniveau ab. Daneben existiert noch ein Cluster von Nichtkäufern, die über das Jahr hinweg keinerlei *Eiscreme* kaufen.
Um zu erkennen, welche unterschiedlichen Einkaufsverläufe den jeweiligen Gruppen bei Clusterverfahren 1 zugeordnet wurden, sind in Abbildung 7.7 50 zufällig ausgewählte Verläufe mit ihrem zugehörigen Zentrum (in rot eingezeichnet) dargestellt. Hieraus wird auch die Problematik deutlich, aus großen Datensätzen mit vielen Verläufen einzelne Struktu-

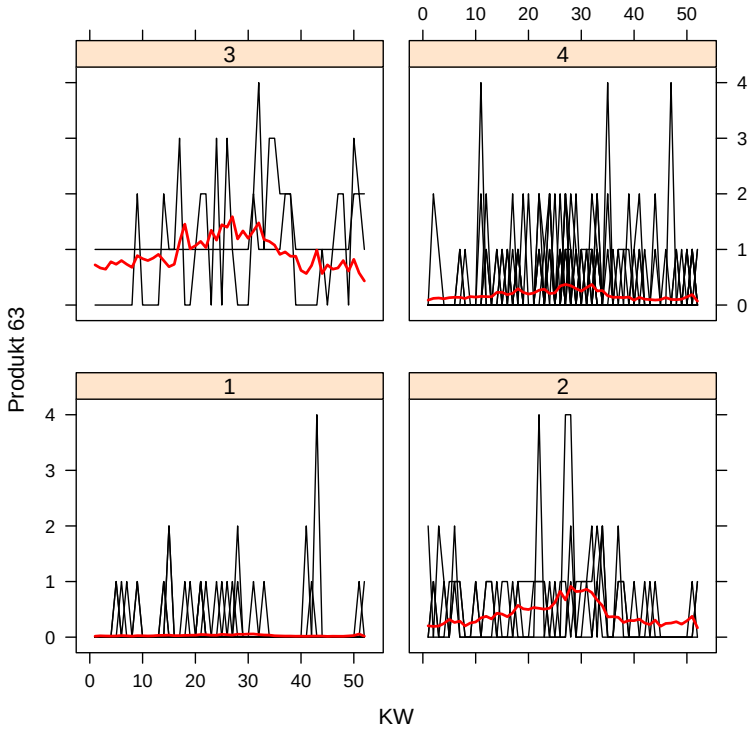


Abbildung 7.7.: Anzahl gekauftes Produkt 63 nach KW, aufgeteilt nach Cluster, Stichprobe von 50 HH – Clustern der Rohdaten mit k-means (k=4) und euklidischer Distanz; Clusterzentrum in rot eingezeichnet.

ren erkennen zu können. So dürfte es eher schwer sein, Clusterverläufe in den einzelnen Gruppen rein optisch voneinander zu trennen.
Auf Monatsebene ergeben sich ähnliche Verläufe der Zentren, allerdings auf etwas höherem Niveau (vgl. Anhang A). Die Basismengen liegen hier bei ca. 0, 1 und 3 Einkäufen und erreichen nach kontinuierlichem Anstieg ihr Maximum im Juli mit ca. 1.5, 3 und 6 monatlichen Käufen. Das Nichtkäufercluster bleibt auch bei monatlicher Betrachtungsweise bestehen.

Clusterverfahren 2 Clusterverfahren 2 mit der Poisson-Distanz anstelle der euklidischen bringt die in Graphik 7.8 dargestellten typischen Transaktionsverläufe für *Eiscreme* auf Kalenderwochenebene hervor.

Rein optisch unterscheiden sich die einzelnen Verläufe zueinander und auch im Hinblick auf Clusterverfahren 1 nur in einer Verschiebung im Niveau und nicht im Verlauf selbst. Ebenfalls erkennbar ist die Nichtkäufer-Gruppe, danach folgen zwei Gruppen mit relativ konstanten Ausgangseinkäufen von 0.1 und 0.2 gekauften Packungen pro Woche und eine etwas höher angesiedelte mit 0.5 Packungen mit jeweils starkem Hoch, mit bis zu 100% Wachstum, im späten Frühjahr bis in den Sommer, analog zu Verfahren 1.

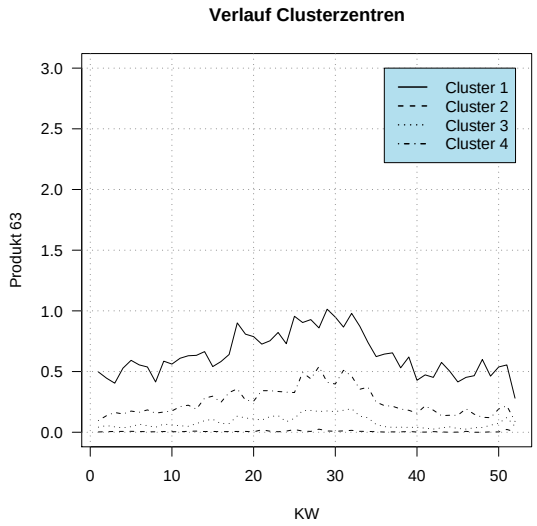


Abbildung 7.8.: Verlauf der Clusterzentren – Clustern der Rohdaten mit k-means und Poisson-Distanz, Produkt 63, KW-Ebene

Die Betrachtung auf Monatsebene bringt auch hier in etwa gleiche Verläufe bei insgesamt höherem Einkaufsvolumen hervor (vgl. Anhang A).

Clusterverfahren 3 Das Standardverfahren beim Clustern von Kurven – Clustern der Splinekoeffizienten mit k-means – liefert im Gegensatz zu Verfahren 1 und 2 teilweise unterschiedliche Transaktionsmuster (vgl. Abbildung 7.9). Hier entsteht sowohl auf Kalenderwochen- als auch auf Monatsebene wiederum ein konstantes Einkaufsverhaltensmuster bei den Nullkäufern. Zusätzlich gibt es nun eine Gruppe von Frühjahrskäufern, die um Mitte Mai herum ihr Maximum von ca. 0.7 Packungen pro Woche bzw. 3 pro Monat erreicht. Darüberhinaus werden zwei Klassen von Sommerkäufern aufgedeckt: Eine relativ stark ausgeprägte Sommergruppe, die um den August herum 1 Packung Eiscreme pro Woche bzw. 5 Packungen pro Monat kauft und eine etwas unterhalb verlaufende Klasse mit wöchentlich maximal 0.3 bzw. monatlich ca. 2 Packungen.

Bei Clusterverfahren 3 lassen sich zudem die an die Transaktionen der Haushalte angepassten Splines, nach ihrer Clusterzuordnung getrennt, abbilden (siehe Abbildung 7.10). Im Gegensatz zu beispielsweise Abbildung 7.7 auf Seite 38 sind hier die verschiedenen Kurvenverläufe wesentlich deutlicher erkennbar. Durch die starke Glättung kristallisieren sich die grundlegenden Unterschiede klarer heraus, als dies beim Plotten der Rohdaten der Fall ist.

Die Splines der Haushalte zeigen beispielsweise in Cluster 3 das markante Frühjahrshoch und in Klasse 1 und 4 den mehr oder weniger deutlich ausgeprägten Anstieg im Hochsommer. In Cluster 2 dagegen sammeln sich die konstanten Kurvenverläufe der Wenig- und Nichtkäufer.

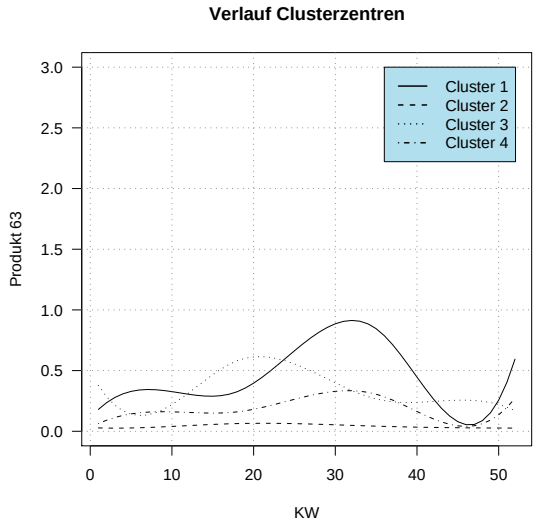


Abbildung 7.9.: Verlauf der Clusterzentren – Clustern der Splinekoeffizienten (B-Splines Grad 3, 2 innere Knoten) mit k-means (k=4) und euklidischer Distanz, Produkt 63, KW-Ebene

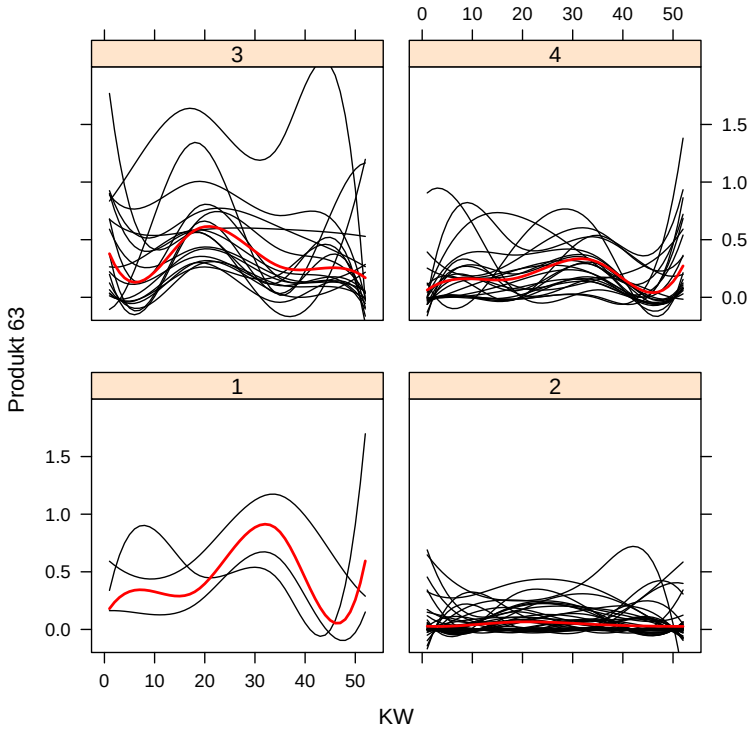


Abbildung 7.10.: Angepasste Splines nach Clusterzuordnung – Clustern der Splinekoeffizienten (B-Splines Grad 3, 2 innere Knoten) mit k-means (k=4) und euklidischer Distanz, Produkt 63 – Stichprobe von 100 HH; Clusterzentren in rot eingezeichnet.

Clusterverfahren 4 Das Partitionieren der mit Hilfe der B-Spline-Expansion angepassten Funktionswerte an den Erhebungszeitpunkten ergibt auf Kalenderwochenebene für Produkt 63 die in Abbildung 7.11 dargestellte Lösung.

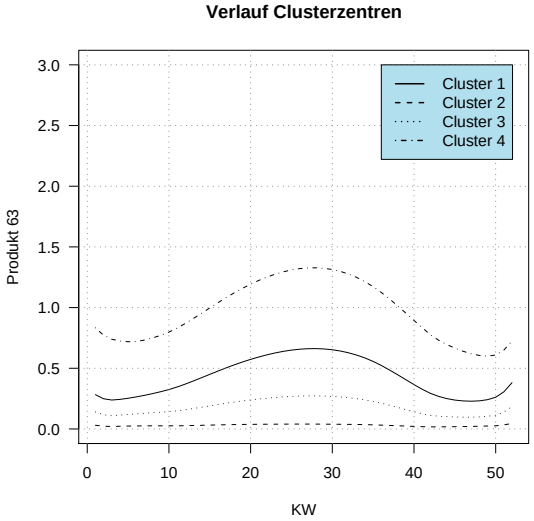


Abbildung 7.11.: Verlauf der Clusterzentren – Clustern der vorhergesagten Werte der Splines (B-Splines Grad 3, 2 innere Knoten) an den Erhebungszeitpunkten mit k-means (k=4) und euklidischer Distanz, Produkt 63, KW-Ebene

Die Zentren verlaufen sowohl auf Wochen- als auch auf Monatsebene (vgl. Anhang A) ähnlich zu denen aus Clusterverfahren 1 – Clustern der Rohdaten mit dem k-means Algorithmus unter Verwendung der euklidischen Distanz – mit dem Unterschied, dass sie geglättet sind. Dies entspricht genau der dahinterliegenden Idee von Verfahren 4 im Gegensatz zu Verfahren 1, der Glättung der Rohdaten aufgrund eines unterstellten funktionalen Verlaufs. So ergibt sich sowohl auf Wochen- als auch auf Monatsebene das klassische Cluster der Nichtkäufer und drei weitere, gleichförmig verlaufende Centroide auf unterschiedlichem Niveau. Diese besitzen eine Basiseinkaufsmenge über den Winter von wöchentlich 0.7 (Cluster 4), 0.2 (Cluster 1) und 0.1 (Cluster 3) Packungen *Eiscreme*, die ab dem Frühjahr bis zum Sommer, Mitte Juli, auf eine Menge von 1.4, 0.7 und 0.2 Packungen ansteigt. Auf Monatsebene liegt das Niveau dieser drei Cluster etwas höher bei ca. 0.5, 1 und 3.5 mit einem Anstieg auf 1, 3 und 6 Packungen pro Monat.

Clusterverfahren 5 Das neue Verfahren des Splineclusterns unter Verwendung der euklidischen Distanz ergibt auf den ersten Blick auf beiden Aggregationsstufen (Kalenderwochen- und Monatsebene) sehr ähnliche Centroide wie Verfahren 4 und auch Verfahren 1. Bei den drei obersten Clusterverläufen in Abbildung 7.12 für die Kalenderwochenebene wird wiederum das Hoch um den Monat Juli herum deutlich. Ansonsten verlaufen diese Zentren fast

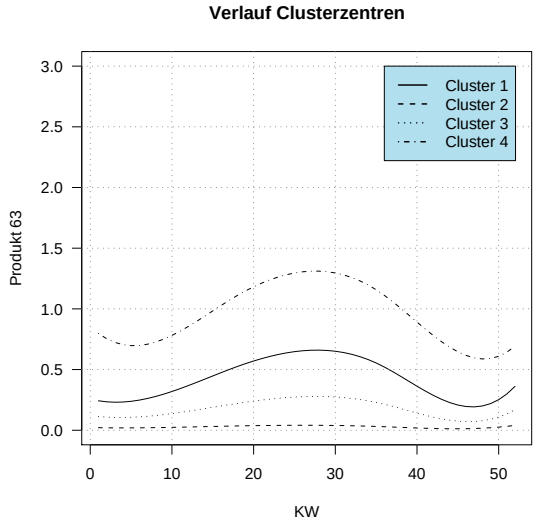


Abbildung 7.12.: Verlauf der Clusterzentren – Clustern der Rohdaten mit Splineclustern (B-Splines Grad 3, 2 innere Knoten, df=6) und euklidischer Distanz (k=4), Produkt 63, KW-Ebene

identisch über das Jahr hinweg auf jeweils unterschiedlichem Basisniveau. Die einzelnen Ausgangs- und Maximaleinkaufsmengen entsprechen den Einkäufen bei Clusterverfahren 4. Bis auf das Nichtkäufercluster steigen die Einkäufe von *Eiscreme* ab Anfang März an, bis sie Mitte Juli mit einer über 100%-igen Steigerung im Vergleich zum Winterniveau ihr Maximum erreichen und danach langsam wieder bis Ende September abfallen (Die Abbildung der Clusterzentren auf Monatsebene befinden sich in Anhang A.).

Clusterverfahren 6 Eine Kombination des Splineclusterns mit der Poissondistanz, das sowohl den funktionalen Verlauf als auch die Verteilung der Daten berücksichtigt, ergibt bei wöchentlicher Betrachtung die Centroide in Graphik 7.13. Hier zeigt sich im Gegensatz zu den Clusterverfahren mit euklidischer Distanz eine etwas „gestauchte“ Clusterlösung. Die stärkeren Käufe in den Sommermonaten bei den oberen zwei Clusterzentren sind in diesem Fall weniger stark ausgeprägt. Vom Niveau entsprechen diese, jeweils die Aggregationsebenen für sich betrachtet, dem Clusterverfahren 2, dem Clustern der Rohdaten mit der neu entwickelten Poisson-Distanz. Das Basisniveau bei den drei neben den Nichtkäufern existierenden Clustern liegt bei ca. 0.1, 0.2 bzw. 0.5 wöchentlichen bzw. 0, 0.5 und 2 monatlichen Käufen (vgl. Anhang A für die Monatsebene) mit ebenfalls einem i.d.R. um bis zu 100% stärkerem Konsum um den Monat Juli herum.

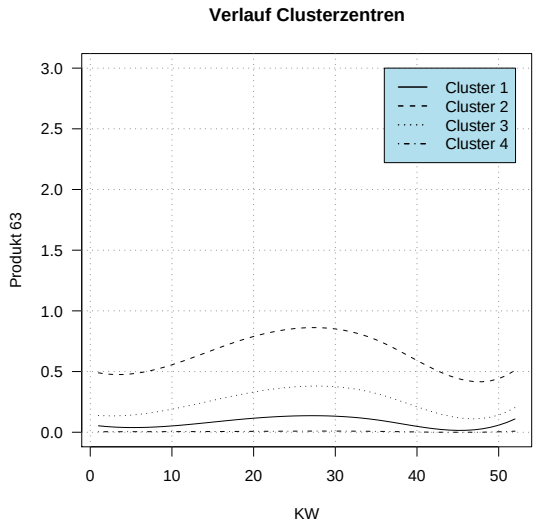


Abbildung 7.13.: Verlauf der Clusterzentren – Clustern der Rohdaten mit Splineclustern (B-Splines Grad 3, 2 innere Knoten, df=6) und Poisson-Distanz (k=4), Produkt 63, KW-Ebene

Vergleich Anhand der Graphiken der Clusterzentren wird bereits deutlich, dass sich die einzelnen Clusterverfahren mehr oder minder stark unterscheiden. So stellt sich die Frage nach einer Maßzahl, die die Ähnlichkeit der Ergebnisse der betrachteten Möglichkeiten miteinander oder mit dem Standardverfahren geeignet vergleichen kann.

Randindex Die bekannteste Methode zwei gegebene Partitionen miteinander zu vergleichen ist der sogenannte *Randindex* (vgl. Rand (1971)). Der Randindex beruht auf dem Vergleich, wie Objektpaare, also jeweils zwei einzelne Beobachtungen, geclustert werden. Dabei wird erfasst, ob die beiden betrachteten Objekte bei zwei Partitionen $\mathscr{C}^{(X)}$ und $\mathscr{C}^{(Y)}$ in jeweils die gleiche Gruppe gelangen oder beispielsweise in Partition $\mathscr{C}^{(X)}$ schon und in $\mathscr{C}^{(Y)}$ nicht. Allgemein differenziert man zwischen Übereinstimmungen und Nichtübereinstimmungen. Als Übereinstimmung zählen Objektpaare, bei denen beide Objekte sowohl in $\mathscr{C}^{(X)}$ als auch in $\mathscr{C}^{(Y)}$ im gleichen oder nicht im gleichen Cluster sind. Dagegen ergeben sich Nichtübereinstimmungen bei Objekten, die in $\mathscr{C}^{(X)}$ im selben und in $\mathscr{C}^{(Y)}$ nicht im selben Cluster verzeichnet werden, oder umgekehrt. Zwei nach dem Randindex ähnliche Partitionen haben eine relativ hohe Anzahl an Übereinstimmungen, während eher unterschiedliche Partitionen eine hohe Anzahl an Nichtübereinstimmungen aufweisen.

Berechnung des Randindex:

(vgl. Rand (1971) und Arabie (1985))

Ausgangspunkt für die Berechnung des Randindex sind die in der Kontingenztafel über die Partitionen $\mathscr{C}^{(X)} = \left\{\mathscr{C}_1^{(X)}, \mathscr{C}_2^{(X)}, \ldots, \mathscr{C}_R^{(X)}\right\}$ und $\mathscr{C}^{(Y)} = \left\{\mathscr{C}_1^{(Y)}, \mathscr{C}_2^{(Y)}, \ldots, \mathscr{C}_S^{(Y)}\right\}$ zusammengefassten Überschneidungen:

	$\mathscr{C}_1^{(Y)}$	$\mathscr{C}_2^{(Y)}$	$\ldots$	$\mathscr{C}_S^{(Y)}$		
$\mathscr{C}_1^{(X)}$	n_{11}	n_{12}	$\ldots$	n_{1S}	n_{1+}	
$\mathscr{C}_2^{(X)}$	n_{21}	n_{22}	$\ldots$	n_{2S}	n_{2+}	
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	
$\mathscr{C}_R^{(X)}$	n_{R1}	n_{R2}	$\ldots$	n_{RS}	n_{R+}	
	n_{+1}	n_{+2}	$\ldots$	n_{+S}	n_{++}	$= n$

mit $n_{ij} = \left|\mathscr{C}_i^{(X)} \cap \mathscr{C}_j^{(Y)}\right|$ als Anzahl gemeinsamer Objekte in Gruppe $\mathscr{C}_i^{(X)}$ und $\mathscr{C}_j^{(Y)}$.

Mit dem Randindex wird die Wahrscheinlichkeit der Übereinstimmung bezeichnet. Diese berechnet sich als Anzahl der Übereinstimmungen relativ zur Anzahl der Übereinstimmungen und Nichtübereinstimmungen.
Die Anzahl der Objekte, die in beiden Partitionen im selben Cluster sind, berechnet sich gemäß

$$\frac{1}{2}\sum_{i=1}^{R}\sum_{j=1}^{S} n_{ij}(n_{ij}-1) \tag{7.1}$$

und die der Objekte, die in beiden Partitionen in unterschiedlichen Clustern sind, nach

$$\frac{1}{2}\left[n^2 + \sum_{i=1}^{R}\sum_{j=1}^{S} n_{ij}^2 - \left(\sum_{i=1}^{R} n_{i+}^2 + \sum_{j=1}^{S} n_{+j}^2\right)\right] \tag{7.2}$$

Die Anzahl der Übereinstimmungen (entspricht der Summe aus 7.1 und 7.2) und Nichtübereinstimmungen berechnet sich als Binomialkoeffizient $\binom{n}{2}$.

Insgesamt ergibt sich für den Randindex (RI):

$$RI = \frac{\binom{n}{2} + \sum_{i=1}^{R}\sum_{j=1}^{S} n_{ij}^2 - \frac{1}{2}\left(\sum_{i=1}^{R} n_{i+}^2 + \sum_{j=1}^{S} n_{+j}^2\right)}{\binom{n}{2}} \tag{7.3}$$

Der Randindex besitzt einen Wertebereich von $RI \in [0,1]$. Er nimmt den Wert 0 an, falls zwei Partitionen keinerlei Ähnlichkeit besitzen und den Wert 1, falls sie vollkommen identisch sind.

Eine Erweiterung des Randindex von Rand (1971) ist der sogenannte *adjusted* Randindex (vgl. Arabie (1985)).[14] Dieser korrigiert den Randindex bezüglich Zufall. Es wird angenommen, dass die vorliegende Kontingenztafel zweier Clusterlösungen einer hypergeometrischen Verteilung entspringt. Dies bedeutet, dass die beiden Partitionen $\mathscr{C}^{(X)}$ und $\mathscr{C}^{(Y)}$ zufällig, unter der Bedingung der echten Anzahl an Klassen und Objekten in jeder Partition, gezogen wurden. Der erwartete Index, unter dieser Annahme, wird nun in die Berechnung miteinbezogen, und der adjusted Randindex ergibt sich als:

Berechnung des adjusted Randindex (aRI):

(vgl. Arabie (1985))

Die allgemeine Formel eines Index mit Zufallskorrektur lautet:

$$aRI = \frac{Index - ExpectedIndex}{MaximumIndex - ExpectedIndex} \tag{7.4}$$

Beim Einsetzen der konkreten Formeln für die Berechnung des Randindex und des erwarteten Index, sowie des maximalen Index ergibt sich:

$$aRI = \frac{\sum_{i=1}^{R}\sum_{j=1}^{S}\binom{n_{ij}}{2} - \sum_{i=1}^{R}\binom{n_{i+}}{2}\sum_{j=1}^{S}\binom{n_{+j}}{2}/\binom{n}{2}}{\frac{1}{2}\left[\sum_{i=1}^{R}\binom{n_{i+}}{2} + \sum_{j=1}^{S}\binom{n_{+j}}{2}\right] - \sum_{i=1}^{R}\binom{n_{i+}}{2}\sum_{j=1}^{S}\binom{n_{+j}}{2}/\binom{n}{2}} \tag{7.5}$$

Auch der *adjusted Randindex* besitzt den Wertebereich $aRI \in [0, 1]$. Beim Wert 0 entspricht der aRI dem erwarteten Index.

In Tabelle 7.1 werden die (adjusted) Randindizes[15] der verschiedenen Verfahren zueinander als Kreuztabelle dargestellt.

	1	2	3	4	5	6
Rohdaten, k-means, eukl - 1	1.00	0.41	0.48	0.88	0.90	0.40
Rohdaten, k-means, poi - 2	0.41	1.00	0.30	0.42	0.41	0.89
Splinekoeffizienten, k-means, eukl - 3	0.48	0.30	1.00	0.48	0.48	0.28
Splinewerte, k-means, eukl - 4	0.88	0.42	0.48	1.00	0.96	0.40
Rohdaten, Splineclustern, eukl - 5	0.90	0.41	0.48	0.96	1.00	0.40
Rohdaten, Splineclustern, poi - 6	0.40	0.89	0.28	0.40	0.40	1.00

Tabelle 7.1.: Vergleich der Clusterlösungen der unterschiedlichen Verfahren, Produkt 63, KW-Ebene, k=4

[14] Der adjusted Randindex ist in der Funktion `Rand` im Package `flexclust` in `R` implementiert.

[15] Im folgenden ist immer mit *Randindex* der *adjusted Randindex* gemeint.

Auffällig sind die hohen Randindizes zwischen Verfahren 1 – Clustern der Rohdaten mit euklidischer Distanz – mit Verfahren 4 – Clustern der Splinewerte mit euklidischer Distanz – (0.88) und Verfahren 5 – Splineclustern mit euklidischer Distanz – (0.90). Somit werden die bereits graphisch deutlichen Ähnlichkeiten beim Centroidverlauf bestätigt. Auch bei den die Poisson-Distanz verwendenden Clustermöglichkeiten 2 – Clustern der Rohdaten – und 6 – Splineclustern – zeigt sich ein hoher Index von 0.89. Ebenso wurden in diesem Fall bereits optische Gemeinsamkeiten der Clusterverläufe festgestellt. Beim Standardverfahren, dem Clustern der Splinekoeffizienten, das bei den Centroiden von allen anderen Verfahren stark abweichende Strukturen aufdeckte, zeigen sich bestätigende Werte des Randindex. So liegt dieser bei einem Vergleich mit den übrigen Clustermethoden nur zwischen 0.28 und 0.48. Dies spricht für eine größere Unähnlichkeit bei der Clusterzuordnung. Ebenfalls bestehen solche Unterschiede beim Vergleich zwischen auf Poisson-Distanz und auf euklidischer Distanz basierenden Methoden. Das Clustern der Rohdaten mit Poisson-Distanz ergibt mit einem Randindex von 0.30 bis 0.42 zu den Verfahren mit klassischem Distanzmaß, gewichtigere Abweichungen in der Clusterzuordnung. Das gleiche Bild zeigt die Gegenüberstellung mit dem Splineclustern unter der Poisson-Distanz (Verfahren 6), hier liegt der Index zwischen 0.28 und 0.40. Diese Auffälligkeiten werden in Abschnitt 7.2.2 näher untersucht.

7.2.1.2. Produkt 61 – Mineralwasser

Die Clusterzentren für die sechs Clusterverfahren für Produkt 61 – *Mineralwasser* auf Monatsebene zeigt Abbildung 7.14. Auffällig ist das bereits von Produktgruppe 63 – *Eiscreme* bekannte *Nullkäufercluster*, das sich über alle Verfahren hinweg ergibt. Auch eine Betrachtung der Transaktionen auf Wochenebene deckt diese Gruppe auf. Weiterhin führen die Clusterlösungen 1, 4 und 5 auch bei Produkt 61 – *Mineralwasser* zu zueinander ähnlichen Zentren. So existiert bei allen diesen drei Clusterverfahren eine Kundengruppe, die sich um die 10 Einkäufe pro Monat bzw. um die 2 Einkäufe pro Woche (für die Ergebnisse auf Kalenderwochenebene vgl. Anhang A), mit einem leichten Sommerhoch in den Monaten Juni bis August, bewegt. Bei den Clusterlösungen mit der Poisson-Distanz (Verfahren 2 und 6) weist dieses Zentrum einen ähnlichen Verlauf auf, allerdings auch – wie bereits bei Produkt 63 – *Eiscreme* festgestellt – auf etwas niedrigerem Gesamtniveau. Hier schwankt die Anzahl an Einkäufen um die 1.2 pro Woche bzw. um die 5 Einkäufe im Monat. Die zwei verbleibenden Zentren verlaufen bei allen Verfahren, außer dem Clustern der Splinekoeffizienten, in etwa parallel. Bei den euklidischen Distanzen ergeben sich Niveaus von 2 und 4 Käufen, bei den Poisson-Distanzen von 1 und 2 auf Monatsebene. Das Clusterverfahren 3 stellt andere Centroide in den mittleren Käufergruppen heraus. Diese kennzeichnen sich in einem gegenläufigen Verlauf, also ein Hoch bei der einen Gruppe entspricht einem Tief bei der anderen. Die eine Gruppe wäre demnach in den Frühjahr- und Herbstmonaten aktiver, die andere in den Sommermonaten.

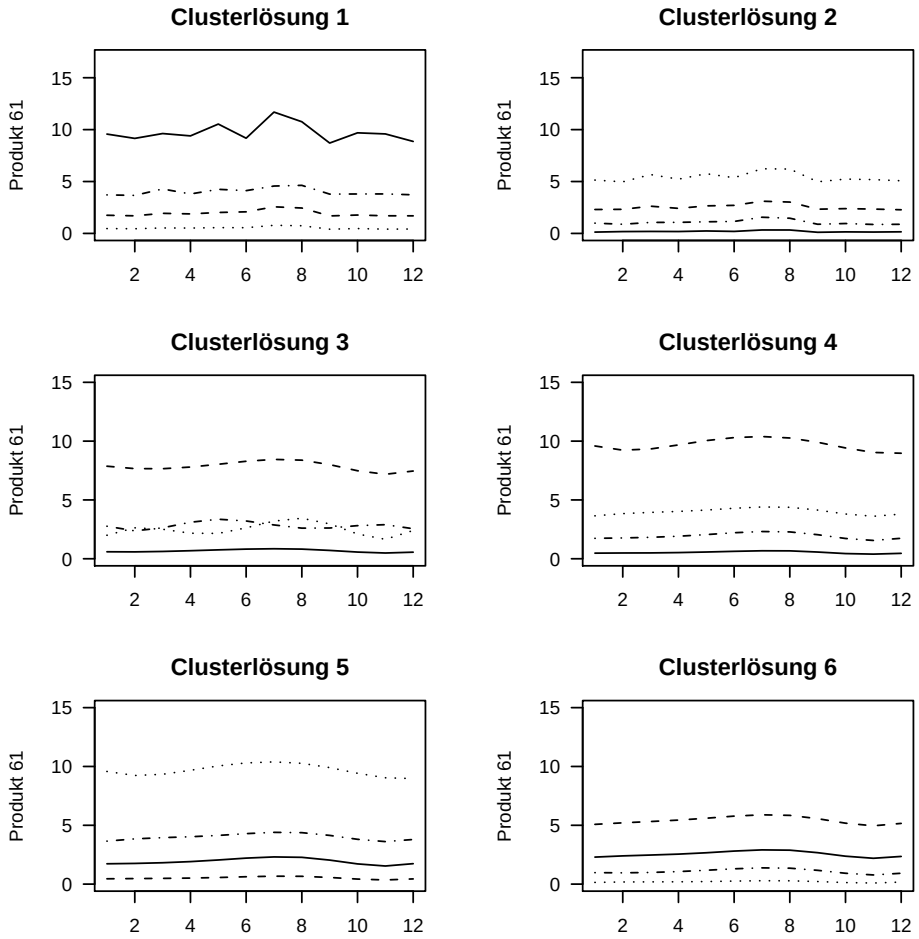


Abbildung 7.14.: Verlauf der Clusterzentren der einzelnen Clusterlösungen, Produkt 61, Monatsebene

7.2.1.3. Produkt 44 – Alkoholfreie Getränke ohne Kohlensäure (Fruchthaltige)

Bei den *alkoholfreien Getränken ohne Kohlensäure* kommt es auf Wochenebene zu den in Graphik 7.15 dargestellten Centroiden.

Die vertraute Gruppe der *Nichtkäufer* ist auch hier auf Kalenderwochen- und Monatsebene über alle Verfahren hinweg vertreten (für die Ergebnisse auf Monatsebene vgl. Anhang A). Augenfällig sind wiederum die Zusammenhänge der Lösungen zwischen den Verfahren 1, 4 und 5 mit der euklidischen Distanz sowie zwischen 2 und 6 mit der Poisson-Distanz. Bei diesen Lösungen existieren wiederum zwei mittlere, relativ parallel und konstant über die Zeit verlaufende Einkaufsmuster, die bei den ersten Verfahren wiederum etwas höher liegen als bei den Verfahren mit der Poisson-Distanz. So liegen beispielsweise die Zentren auf Monatsebene im ersten Fall bei ca. 3 und 6 Einkäufen pro Monat, während sie im zweiten Fall bei 1 und 3.5 liegen. Das obere Cluster, das meist mehr Struktur aufweist, zeigt auch bei diesem Produkt den markantesten Verlauf. Bei den *alkoholfreien Geträn-*

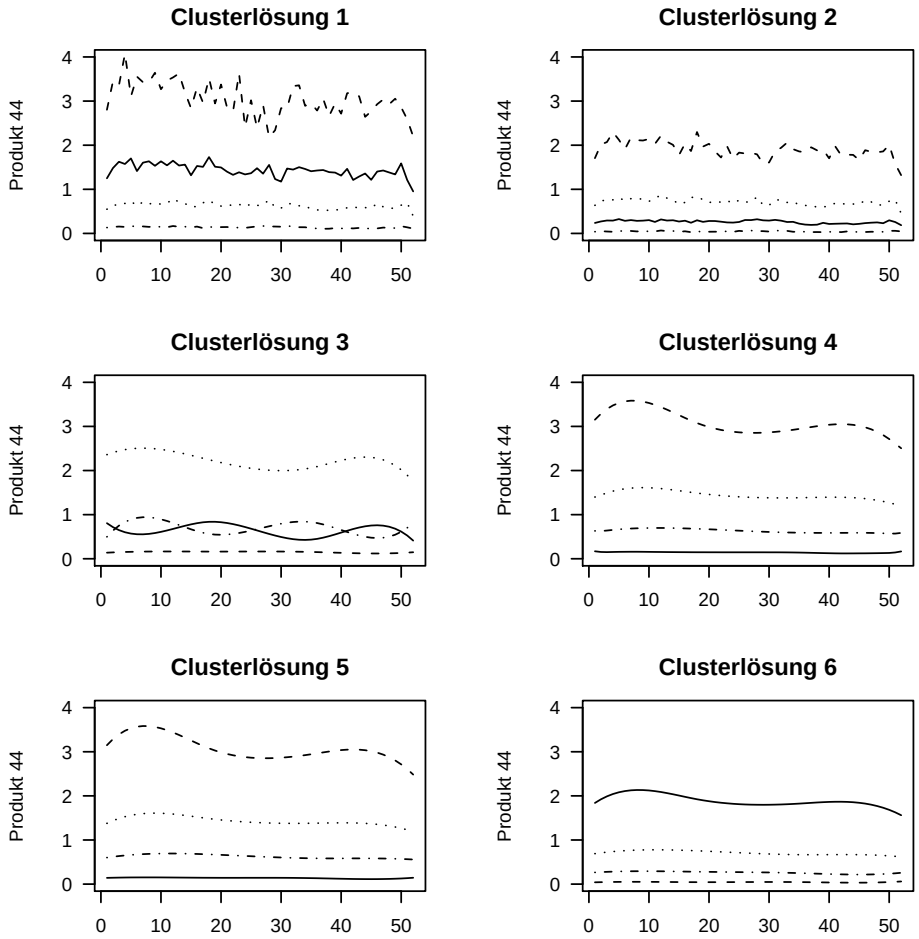


Abbildung 7.15.: Verlauf der Clusterzentren der einzelnen Clusterlösungen, Produkt 44, KW-Ebene

ken ohne Kohlensäure verzeichnet sich ein ausgeprägtes Frühjahrshoch und ein weiterer kleiner Anstieg in den Transaktionen im Frühsommer zwischen Mitte Mai und Mitte Juli. Diese Hochpunkte nivellieren sich etwas bei den Clusterverfahren 2 und 6, sowie bei einer groberen Betrachtungsweise auf Monatsebene.

7.2.1.4. Produkt 19 – Zahnpasta

Der Verlauf der Zentren auf Monatsebene bei der als relativ konstant bezüglich des Kaufs eingeschätzten *Zahnpasta* (Produkt 19) ist Abbildung 7.16 zu entnehmen. Markant ist hier, dass die einzelnen typischen Transaktionsverläufe sehr flach verlaufen und auf Monatsebene nur zwischen keinem und maximal 4 Einkäufen pro Monat differenzieren. Das Cluster mit den *Nullkäufern* ist wiederum in allen Lösungen über beide Aggregationsstufen hinweg vorhanden. Daneben sind zwei mittlere Gruppen, die beide um einen Einkauf pro Monat schwanken und eine letzte Gruppe, die sich bei Verfahren 1, 4, und 5 bei 4 Zahnpastakäu-

fen pro Monat bzw. bei den restlichen Verfahren bei ca. 3 bewegt. Auch in diesem Fall ist das gegenläufige Verhaltensmuster der beiden mittleren Centroide beim Clustern der Splinekoeffizienten in Verfahren 3 hervorstechend. Auf Kalenderwochenebene lassen sich kaum noch sinnvolle Unterschiede zwischen dem Einkaufsverhalten erkennen, da die einzelnen Gruppen alle nur keines oder ein Produkt in der Woche kaufen (für die Ergebnisse auf Kalenderwochenebene vgl. Anhang A).

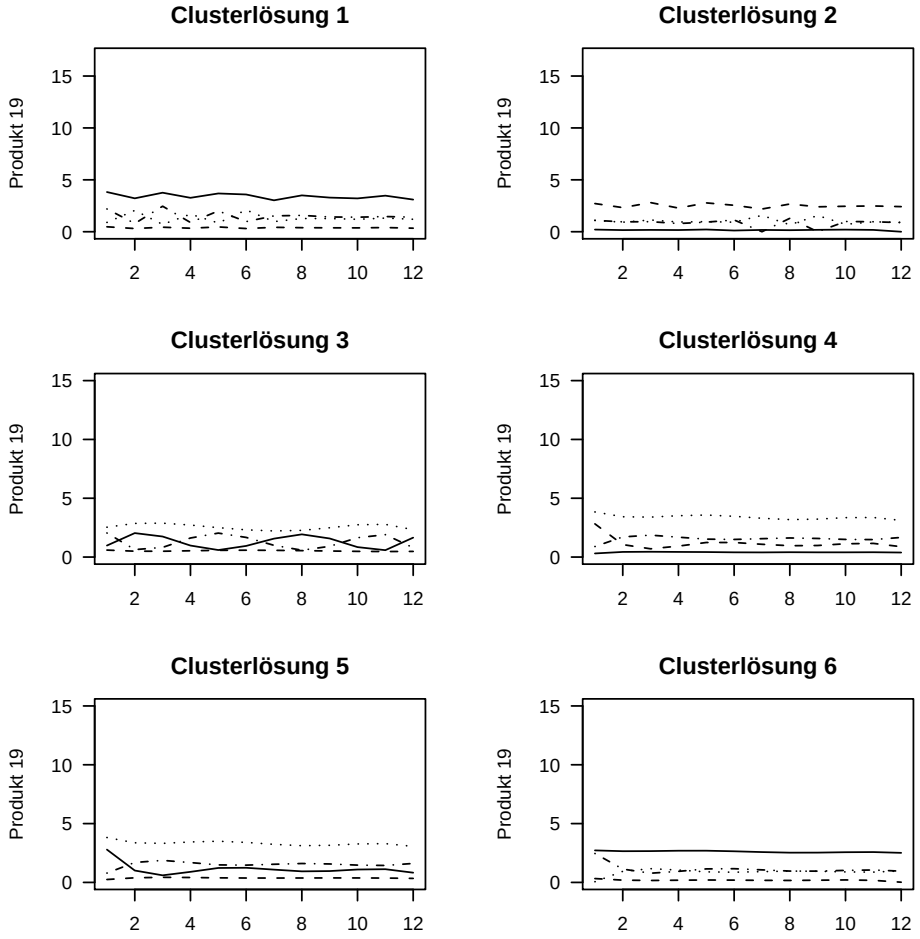


Abbildung 7.16.: Verlauf der Clusterzentren der einzelnen Clusterlösungen, Produkt 19, Monatsebene

7.2.2. Vergleich der Clusterlösungen der Transaktionsdaten

Bereits aus Tabelle 7.1 auf Seite 45 wurde klar, dass die einzelnen betrachteten Clusterverfahren mehr oder weniger starke Abweichungen bei der Klasseneinteilung der Haushalte besitzen. Interessant ist hier, ob die Ähnlichkeiten der Verfahren zueinander über alle betrachteten Produkte und auch über die beiden Aggregationsniveaus, also sowohl auf

Kalenderwochenebene als auch auf Monatsebene, gegeben sind. Anhand der Verläufe der einzelnen Clusterzentren waren bereits Auffälligkeiten entdeckt worden. So ist eine Vermutung, dass sich das Clustern der Rohdaten mit der euklidischen Distanz (Verfahren 1) nicht sehr stark von Clusterverfahren 5, dem Splineclustern mit der euklidischen Distanz unterscheidet. Aber auch zwischen dem Partitionieren der vorhergesagten Splinewerte an den Erhebungszeitpunkten in Verfahren 4 und dem Splineclustern scheint es Zusammenhänge zu geben. An dieser Stelle ist zudem interessant, wie sich die neue Poisson-Distanz und das Splineclustern als funktionale Herangehensweise gegenüber der herkömmlichen Distanz, sowie ohne Unterstellung eines stetigen Verlaufs verhalten. Aus den Kreuztabellen der Randindizes (siehe Tabelle 7.1, sowie Anhang B) sind folgende Randindizes zu entnehmen:

Clusterverfahren i − j	Produkt Aggregationsniveau	63	61	44	19
1 − 2	KW	0.41	0.33	0.45	0.26
	Monat	0.40	0.37	0.47	0.21
1 − 5	KW	0.90	0.97	0.97	0.54
	Monat	0.93	0.98	0.98	0.59
4 − 5	KW	0.96	0.99	0.98	0.58
	Monat	0.96	0.98	0.99	0.97
2 − 6	KW	0.89	0.82	0.98	0.37
	Monat	0.93	0.96	0.99	0.49
5 − 6	KW	0.40	0.37	0.45	0.47
	Monat	0.40	0.36	0.46	0.41

Tabelle 7.2.: Randindex zwischen Clusterverfahren i und Clusterverfahren j aufgeteilt nach Produkt und Aggregationsniveau

Verfahren 4 − 5: Unterschied Clustern der Splinewerte und Splineclustern bei euklidischer Distanz Tatsächlich lassen sich, wie vermutet, die größten Randindizes zwischen Clusterverfahren 4 und 5, also Clustern der Splinewerte und Splineclustern mit euklidischer Distanz feststellen (vgl. Tabelle 7.2). Einzig auffällig ist hier die Betrachtung des Produktes 19 auf Kalenderwochenebene. Dort liegt der Index beim Vergleich dieser beiden Verfahren auf wöchentlicher Basis zwar immer noch am höchsten, ist aber generell mit 0.58

niedrig. Produkt 19 – *Zahnpasta* hat insgesamt ein sehr niedrig verlaufendes Einkaufsmuster. Oft wird keine oder nur eine Packung Zahnpasta in der Woche oder sogar im Monat gekauft. Dadurch entstehen Transaktionsverläufe, die sich in ihrem Muster kaum unterscheiden und nur geringe bis gar keine Strukturen aufweisen. Das Produkt 19 dient damit der Untersuchung des Verhaltens der Clusterverfahren bei wenig Strukturunterschieden. Die Unterschiede zwischen den Verfahren bei diesen Daten sind insgesamt – wie es sich in den niedrigen Randindizes widerspiegelt – größer.

Verfahren 1 – 5 und 2 – 6: Unterschied Splineclustern und Clustern der Rohdaten Tabelle 7.2 zeigt außerdem, dass auch das Clustern der Rohdaten mit der euklidischen Distanz (Verfahren 1) starke Überschneidungen hinsichtlich der Klasseneinteilung mit dem Splineclustern (Verfahren 5) hat. Hier schwanken, bis auf die strukturarme *Zahnpasta*, die Randindizes zwischen 0.90 und 0.98, also zwischen extrem ähnlicher bis nahezu identischer Gruppenbildung. Insgesamt ist der Zusammenhang in diesem Fall nicht ganz so hoch wie zwischen Verfahren 4 und 5.
Der Unterschied zwischen der funktionalen Herangehensweise beim Splineclustern auf der einen Seite und einem Ignorieren dieser funktionalen Form beim Clustern der Rohdaten auf der anderen Seite, lässt sich bei einem zusätzlichen Vergleich von Verfahren 2 mit Verfahren 6 beurteilen. So zeigt sich bei den strukturstärkeren Produkten 63, 61 und 44 große Ähnlichkeit zwischen dem Clustern der Rohdaten mit der neuen Poisson-Distanz und dem Splineclustern mit dieser. Der Randindex variiert hier von 0.82 bis sogar 0.99, also fast vollkommen identischer Zuordnung. Die Beachtung des glatten Kurvenverlaufs der Transaktionen durch das Splineclustern scheint somit bei den realen Daten, sowohl bei Verwendung der euklidischen als auch der Poisson-Distanz, keine größeren Abweichungen bei der Klasseneinteilung hervorzubringen.

Verfahren 1 – 2 und 5 – 6: Unterschied im Distanzmaß Bei Betrachtung der Randindizes, die den Zusammenhang der Clusterlösungen 1 mit 2 bzw. 5 mit 6, also die Clusterverfahren ohne Berücksichtigung der funktionalen Struktur der Daten und, als Gegenstück, dem Splineclustern mit der Berücksichtigung, zwischen den beiden Distanzmaßen widergeben, lässt sich ein deutlicher Unterschied erkennen. Anders als beim Vergleich der Verfahren innerhalb gleicher Distanzen, führt ein unterschiedliches Distanzmaß zu weit größeren Differenzen in der Klassenbildung. Die Indizes liegen hierbei nie über 0.47.

Verfahren 3: Unterschied zum Standardverfahren Abschließend soll kurz auf das Standardverfahren, dem Clustern von Splinekoeffizienten mit dem k-means Algorithmus, eingegangen werden. Bereits anhand der Verläufe der Clusterzentren ließen sich größere Abweichungen bei oft zwei von vier völlig anderes verlaufenden Centroiden feststellen. Auf Wochenebene erreicht der Randindex des Verfahrens 3 zu je einer aller anderen Cluster-

möglichkeiten einen Wert zwischen 0.09 und 0.56. Auch bei der monatlichen Betrachtung liegt der Index nur zwischen 0.15 und 0.58 (vgl. Tabellen in Anhang B). Die Zuordnung der Haushalte in Klassen unterscheidet sich demnach beim Clustern der Splinekoeffizienten meist sehr stark.

An dieser Stelle ist festzuhalten, dass bei den realen Transaktionsdaten die unterschiedlichen Klassenzuordnungen der einzelnen Verfahren eher durch ein unterschiedliches Distanzmaß, als durch das Verwenden des Splineclusterns im Gegensatz zum Clustern der Rohdaten zustandekommen. Eine Aussage, welches Verfahren am besten für die Daten geeignet ist, lässt sich aber aufgrund der unbekannten Klassen nicht feststellen. Ob nun beispielsweise das Standardverfahren die anderen Verfahren hinsichtlich der richtigen Gruppenbildung übertrifft oder welches Distanzmaß hier sinnvoller ist, kann nur auf andere Art geklärt werden.

7.3. Simulierte Daten

Um nun nicht nur die Ähnlichkeit der Verfahren zueinander, sondern auch deren Leistung in Hinsicht auf das Auffinden der wahren Klassenzugehörigkeit beurteilen zu können, müssen die einzelnen Gruppen im Vorfeld des Clusterns bekannt sein. Aus diesem Grund wurden für diese Fragestellung Daten simuliert. Wie bereits in Abschnitt 3.1.2 erwähnt, ist der Splinegrad und auch die Anzahl und die Orte der gesetzten Knoten sehr entscheidend für die Funktionsanpassung. Beim Testen der einzelnen Clusterlösungen sollten diejenigen Möglichkeiten, die eine Splineinterpolation beinhalten, den Vorteil erhalten, denselben Splinegrad und dieselbe Knotenmenge zu verwenden, die in den Daten steckt. Also sollten nur Kurven geclustert werden, von denen auch die Extrema durch diese Vorgaben ausreichend dargestellt werden können. Um dies zu erreichen, wurden für die Simulation die Clusterergebnisse für das Clustern der Splinekoeffizienten mit dem k-means-Algorithmus (Verfahren 3) aus der vorhergehenden Analyse verwendet. Die Splineexpansionen in den entsprechenden Verfahren sind damit mit den simulierten Daten hinsichtlich der Form, die sie darstellen können, abgestimmt. Bei einer solchen Herangehensweise wird die größtmögliche Aufdeckung der wahren Cluster erwartet. Entsprechen Splinegrad und Knoten dagegen nicht den Vorgaben, ist es für das Verfahren aufgrund mangelnder oder auch zu großer Flexibilität in der Splineanpassung sicher schwerer, die richtigen Klassen zu finden.

Abbildung 7.17 verdeutlicht dies am Beispiel des Clusterns von Splinekoeffizienten bei poissonverteilt simulierten Daten auf Basis von Produkt 63 – *Eiscreme*. Dadurch, dass die simulierten Daten auf denselben Vorgaben beruhen, wie das Clustern selbst, entsprechen

sich die Kurvenverläufe der Splineexpansionen der simulierten Daten und der Kurvenverlauf der Clusterzentren viel mehr, als dies z.B. bei den realen Daten in Abbildung 7.10 der Fall ist.

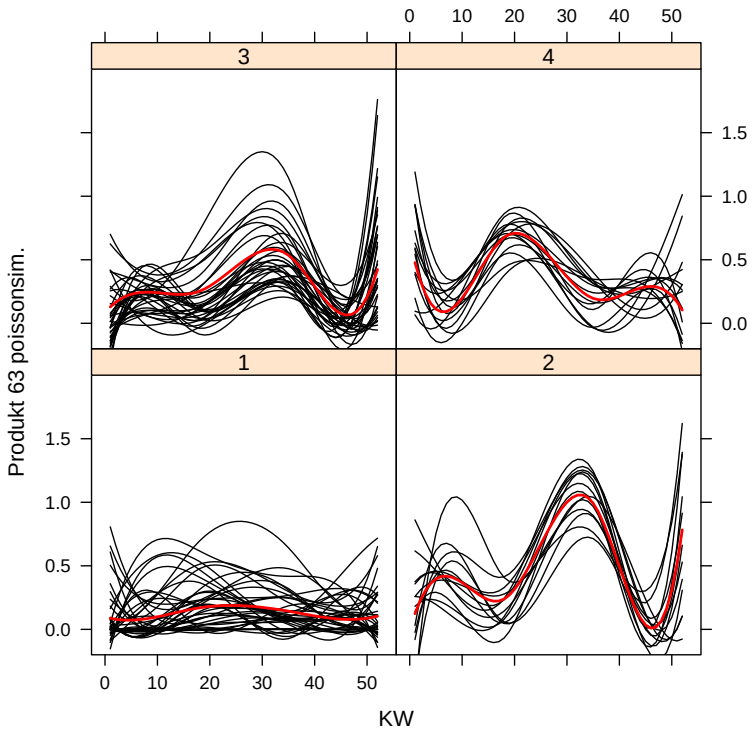


Abbildung 7.17.: angepasste Splines nach Clusterzuordnung – Clustern der Splinekoeffizienten (B-Splines Grad 3, 2 innere Knoten) mit k-means (k=4) und euklidischer Distanz, poissonverteilt simulierte Daten, Produkt 63, Stichprobe von 100 HH

Bei den simulierten Daten ist es nun möglich eine eher den Daten angemessene Splineexpansion zu verwenden, da die Muster der Simulationsbasis entsprechend kurvig verlaufen.

In Abbildung 7.18 sind die Verläufe der Clustercentroide, die nun die Grundlage für die Simulation stellen, für alle vier Produktgruppen auf der Kalenderwochenebene aufgezeichnet. Die Verläufe auf Monatsebene finden sich in Anhang C. Die Clusterzentren für Produkt 61 – *Mineralwasser* und Produkt 44 – *Alkoholfreie Getränke ohne Kohlensäure (Fruchthaltige)* ähneln sich dabei im Verlauf. Es gibt jeweils ein Zentrum, das relativ nahe bei 0 verläuft und ein Zentrum mit einem deutlich höheren Verlauf. Bei Letzterem bewegt sich die Einkaufsmenge auf Kalenderwochenebene bei beiden Produkten um 2 Stück und auf Monatsebene bei ca. 10 bzw. ca. 7 Käufen. Zwei weitere Centroide verlaufen auf etwa gleichem Niveau, aber wechseln sich dabei in der Schwankung ab. Während der Transaktionsverlauf bei einem Zentrum am Messzeitpunkt hoch ist, ist dieser beim anderen niedrig.

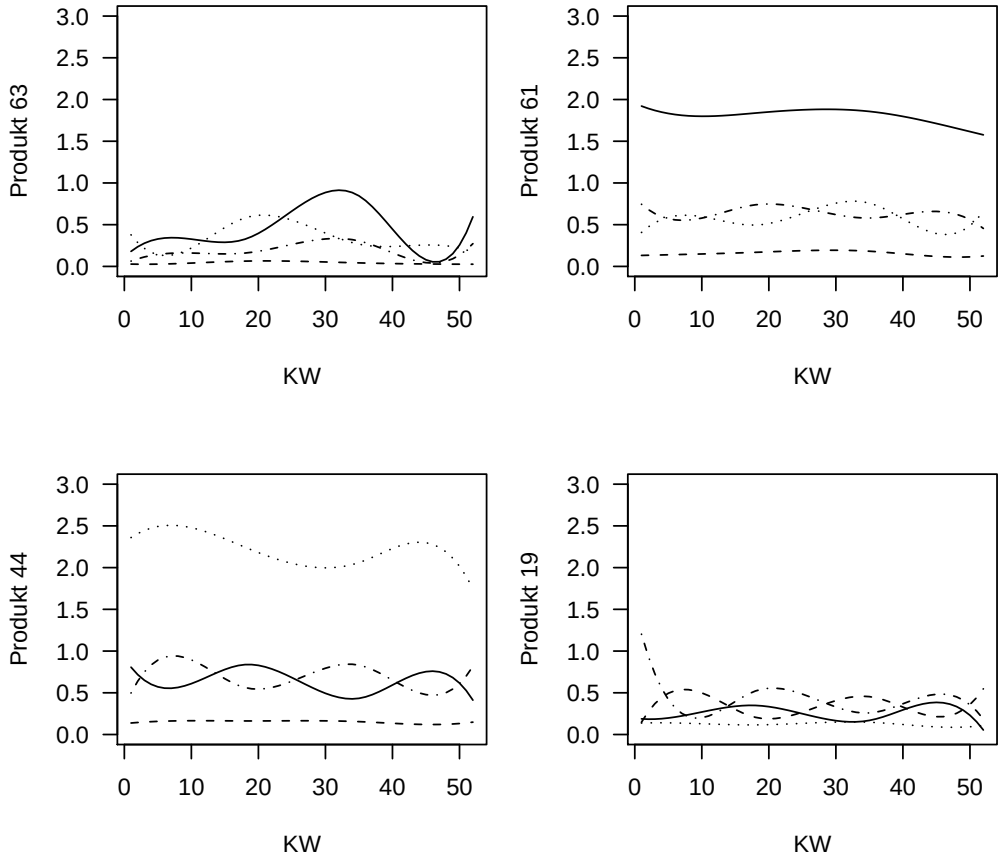


Abbildung 7.18.: Verlauf der Clusterzentren der als Basis dienenden Cluster für die simulierten Daten, KW-Ebene

Diese Schwankungen liegen in einem Fenster von ungefähr 0.5 Einkäufen auf Kalenderwochenebene und ca. einem bei monatlicher Betrachtung. Produkt 63 weist ebenfalls einen Clusterverlauf nahe bei 0 auf, zwei Zentren, die abwechselnd schwanken und eines, dass sich auf etwas höherem Niveau ungefähr gleichläufig zu einem der beiden mittleren Cluster bewegt. Insgesamt finden die Bewegungen im Einkaufsverhalten in einem kleinen Bereich von 0 bis 1 Einkauf auf Kalenderwochenebene, sowie zwischen 0 und 5 Einkäufen auf Monatsebene, statt. Produkt 19 weist keine großen Unterschiede in der Anzahl der Einkäufe auf. Die Zentren bewegen sich alle nur in einem Spektrum von 0 bis 0.5 Einkäufe pro Woche bzw. 0 bis ca. 2.5 pro Monat. Es existieren dennoch zwei relativ konstant laufende Centroide, einer am unteren Rand und einer am oberen Rand. Die beiden mittleren Cluster schwanken wiederum abwechselnd. Trotz des unterschiedlichen Verlaufs, lässt sich aufgrund der geringen Schwankungsbreite, diese Simulationsbasis als strukturarm bezeichnen und dient dazu, das Verhalten der Clusterlösungen bei solchen Daten zu untersuchen.

poissonverteilte Daten Für die Simulation wurden die vorhergesagten Einkaufszahlen an den Messzeitpunkten jeweils als unabhängig poissonverteilt angesehen und für jeden Zeitpunkt Beobachtungen aus einer Poissonverteilung mit Parameter $\lambda = c^{(k)}(t_j)$ gleich dem $k - ten$ Centroid zu dem entsprechenden Zeitpunkt t_j gezogen.[16] Die Clusteranzahl entspricht somit der Anzahl der Cluster in der Basis, also $k = 4$. Noch festzulegen war dagegen die Anzahl der Beobachtungen selbst. Hierzu wurde der Randindex zwischen den einzelnen Verfahren und der wahren Clusterzugehörigkeit nach der Anzahl an simulierten Beobachtungen berechnet, um herauszufinden, wie sich die Klassenzuordnung mit von 50 auf 500 pro Cluster ansteigender Zahl an simulierten Beobachtungen generell verhält. Tabelle 7.3 beinhaltet exemplarisch die Randindizes für Produkt 61 – *Mineralwasser* auf Kalenderwochenebene. Bild 7.19 zeigt diese Entwicklung noch graphisch dargestellt. [17]

	50	100	150	200	250	300	350	400	450	500
1	0.59	0.62	0.62	0.62	0.61	0.69	0.62	0.61	0.68	0.70
2	0.67	0.73	0.68	0.71	0.73	0.73	0.74	0.73	0.70	0.73
3	0.35	0.36	0.30	0.34	0.33	0.32	0.35	0.33	0.31	0.33
4	0.71	0.76	0.73	0.74	0.78	0.75	0.77	0.74	0.74	0.76
5	0.71	0.76	0.75	0.74	0.61	0.75	0.77	0.74	0.74	0.62
6	0.73	0.77	0.73	0.76	0.80	0.75	0.76	0.75	0.76	0.77

Tabelle 7.3.: Randindex zwischen wahrer Clusterzugehörigkeit und der entsprechenden Clusterlösung nach Anzahl simulierter Beobachtungen pro Cluster, poissonverteilt simulierte Daten, Produkt 61, KW-Ebene

Die Randindizes verändern sich bei Zunahme an simulierten Beobachtungen meist nicht sehr stark. Die Verfahren bleiben bzgl. der richtigen Klassenzuordnung auch bei steigender Anzahl an Beobachtungen relativ konstant. Es scheint, als ob die Ergebnisse der einzelnen Verfahren nicht so stark von der Anzahl der simulierten Daten abhängen. Letztendlich wurden für die weiteren Analysen 500 Beobachtungen pro Cluster und somit 2000 Kurvenverläufe insgesamt als Anzahl simulierter Daten festgelegt.

[16] Die Funktion zur Simulation der poissonverteilten Daten mit entsprechender Klassenzuordnung ist in einer bisher unveröffentlichen Erweiterung des `flexclust` Paketes als Funktion `expPoisson` implementiert, ©Friedrich Leisch

[17] Die Tabellen und Graphiken für alle Produkte und Aggregationsebenen finden sich in Anhang F.

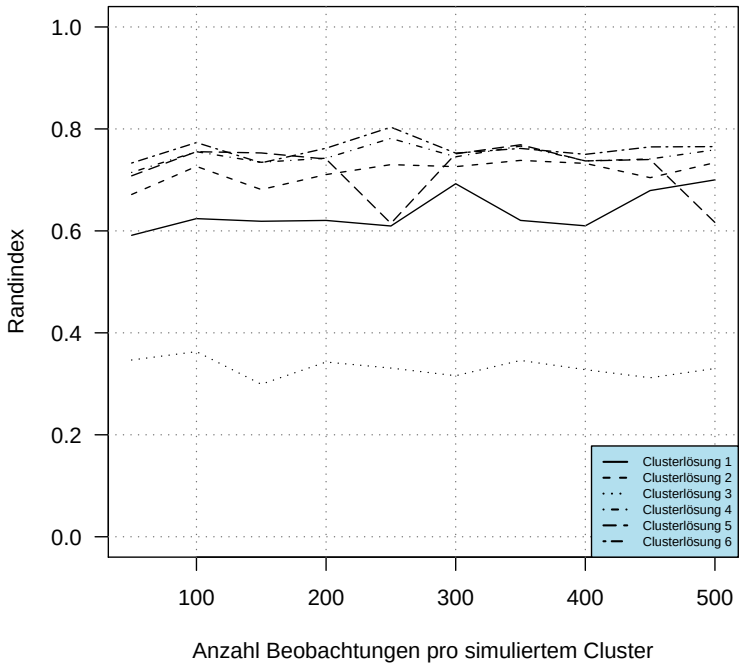


Abbildung 7.19.: Verlauf des Randindex für die einzelnen Clusterlösungen nach steigender Anzahl simulierter Beobachtungen – poissonverteilt simulierte Daten, Produkt 61, KW-Ebene

normalverteilte Daten Zusätzlich zu poissonverteilten Daten wurden auch normalverteilte Daten mit Hinblick auf das Verhalten des neu entwickelten Splineclusterns aus Abschnitt 6.2 untersucht. Bei der Simulation wurden, analog zur poissonverteilten Vorgehensweise, die Beobachtungen zu den Zeitpunkten als unabhängig gesehen und aus einer Normalverteilung mit $\mu = c^{(k)}(t_j)$ gleich dem $k - ten$ Centroid zu dem entsprechenden Zeitpunkt t_j und einer festen Varianz $\sigma^2 = 1$ gezogen.

7.3.1. Verläufe der Clusterzentren mit Vergleich

7.3.1.1. poissonverteilt simulierte Daten

Auch bei den poissonverteilt simulierten Kurven wurden alle sechs vorgestellten Möglichkeiten des Clusterns (siehe Box auf Seite 32) durchgeführt. Wie schon bei den realen Transaktionsdaten ähneln sich einige Clusterzentrenverläufe von einzelnen Verfahren mehr als andere. Abbildung 7.20 zeigt exemplarisch die Clusterlösungen für die poissonverteilt simulierten Daten nach der Vorlage von Produkt 61 – *Mineralwasser* auf Wochenebene.[18]

[18]Für die Clusterlösungen aller Produkte bei poissonverteilt simulierten Daten siehe Anhang D.

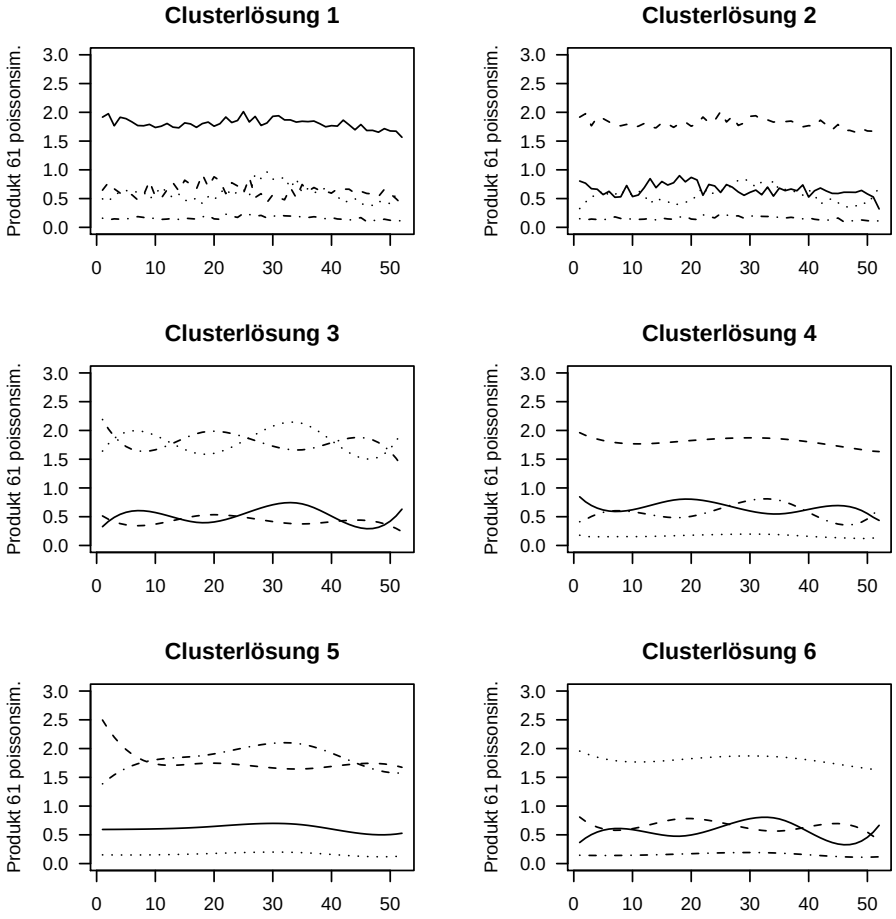


Abbildung 7.20.: Verlauf der Clusterzentren für die einzelnen Clusterlösungen, Produkt 61, poissonverteilt simulierte Daten, KW-Ebene

Deutlich erkennbar sind Zusammenhänge zwischen Verfahren 1, 2, 4 und 6. Anders als vorher ergibt Clusterlösung 5 ein differentes Bild zu dem meist ähnlich verlaufenden Muster beim Clustern der Rohdaten mit euklidischer Distanz (Verfahren 1). Dagegen bleibt der bekannte Unterschied des Standardverfahrens (Lösung 3) mit allen anderen Verfahren auch bei den poissonverteilt simulierten Daten bestehen.
Eine Beurteilung der Zusammenhänge anhand des Randindex ist in Tabelle 7.4 aufgeführt. Klar ersichtlich sind die höheren Indizes zwischen den Verfahren 1 und 2, 4 und 6, sowie die zwischen 2 und 4 bzw. 2 und 6. Das Splineclustern mit der euklidischen Distanz (Verfahren 5) hat bei diesen Daten nur noch Indizes bis zu 0.63 im Vergleich zu anderen Verfahren. Die Ähnlichkeiten zwischen Verfahren mit gleichen Distanzmaßen, wie sie bei den realen Transaktionsdaten in Abschnitt 7.2.1.1 auf Seite 43 festgestellt wurden, kann bei den hier simulierten Daten nicht bestätigt werden. Die Frage, die sich stellt, ist, ob bei den poissonverteilt simulierten Daten insgesamt eventuell andere oder auch gleiche Ähnlichkeiten, wie bei den realen Daten, entdeckt werden können. So sollte noch eine weitere Lösung

betrachtet werden, beispielsweise die für das Produkt 63 in Abbildung 7.21.

	1	2	3	4	5	6
Rohdaten, k-means, eukl - 1	1.00	0.79	0.34	0.76	0.63	0.75
Rohdaten, k-menas, poi - 2	0.79	1.00	0.34	0.81	0.62	0.82
Splinekoeffizienten, k-means, eukl - 3	0.34	0.34	1.00	0.39	0.31	0.40
Splinewerte, k-means, eukl - 4	0.76	0.81	0.39	1.00	0.63	0.92
Rohdaten, Splineclustern, eukl - 5	0.63	0.62	0.31	0.63	1.00	0.62
Rohdaten, Splineclustern, poi - 6	0.75	0.82	0.40	0.92	0.62	1.00

Tabelle 7.4.: Randindex zwischen den verschiedenen Clusterlösungen, n=500, poissonverteilt simulierte Daten, Produkt 61, KW-Ebene

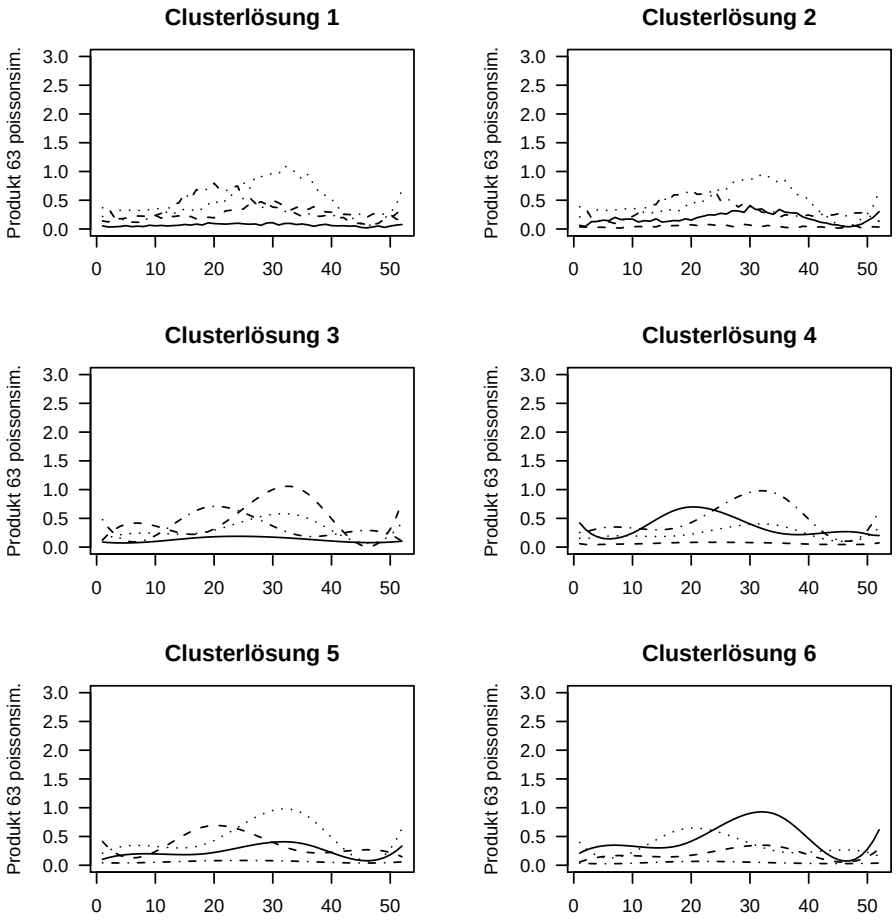


Abbildung 7.21.: Verlauf der Clusterzentren für die einzelnen Clusterlösungen, Produkt 63, poissonverteilt simulierte Daten, KW-Ebene

Nach rein optischen Gesichtspunkten erscheinen bei diesen simulierten Daten alle Verfahren bis auf das Standardverfahren (Verfahren 3) sehr ähnlich zu sein (vgl. Abbildung 7.21). Aber wie wird der Zusammenhang der einzelnen Methoden durch den Randindex erklärt? Um für alle betrachteten Produktgruppenbasen die Ähnlichkeit zueinander beurteilen zu können, wurden wiederum die Randindizes einzelner Verfahren zueinander, analog zu Tabelle 7.2 für die echten Transaktionsdaten, zusammengefasst (siehe Tabelle 7.5).

Clusterverfahren i − j	Produkt / Aggregationsniveau	63	61	44	19
1 − 2	KW	0.58	0.79	0.93	0.62
	Monat	0.73	0.90	0.62	0.72
1 − 5	KW	0.82	0.63	0.93	0.84
	Monat	0.97	0.94	0.97	0.97
4 − 5	KW	0.97	0.63	0.99	0.97
	Monat	0.98	0.99	1.00	0.98
2 − 6	KW	0.94	0.82	0.92	0.86
	Monat	0.96	0.97	0.63	0.94
5 − 6	KW	0.68	0.62	0.95	0.64
	Monat	0.77	0.90	0.95	0.74

Tabelle 7.5.: Randindex zwischen Clusterverfahren i und Clusterverfahren j aufgeteilt nach Produkt und Aggregationsniveau, poissonverteilt simulierte Daten

Verfahren 3: Unterschied zum Standardverfahren und Auffälligkeiten im Aggregationsniveau Analog zu den realen Transaktionsdaten erreicht auch hier das Standardverfahren 3 im Vergleich zu allen anderen Clustermöglichkeiten keine hohen Randindizes, sie liegen zwischen 0.34 und 0.62 (vgl. Anhang E). Dagegen zeigen sich insgesamt bei den poissonverteilt simulierten Daten der strukturstärkeren Produkte (Produkte 63, 61 und 44), anders als zuvor, zwischen den einzelnen Aggregationsebenen selbst größere Abweichungen. Der Vergleich zweier Verfahren auf Wochen- oder Monatsebene ist nicht immer gleich. Um die wahren Auswirkungen einer gröberen Betrachtungsweise beurteilen zu können, wurden hier die verschiedenen Ebenen getrennt voneinander auf den Kalenderwochen- und Monatsbasen simuliert und die monatliche Stufe stellt nicht, wie zuvor, nur eine Aggregation der Wochendaten dar.

Verfahren 4 – 5 sowie 1 – 5 und 2 – 6: Unterschied Splineclustern und Clustern der Splinewerte bzw. der Rohdaten Auf Monatsebene besteht immer noch über alle Produkte hinweg ein großer Zusammenhang zwischen dem Splineclustern mit euklidischer Distanz und dem Clustern der angepassten Splines (5 – 4), sowie zum Clustern der Rohdaten (5 – 1) – jeweils mit euklidischer Distanz. Die beiden splinebasierten Verfahren ergeben auch hier fast identische Klasseneinteilungen. Weiterhin ist die auffallend große Ähnlichkeit beider Verfahren mit Poissondistanz (2 – 6) zueinander nicht so eindeutig, wie bei den echten Transaktionsdaten, aber meist immer noch gegeben.

Verfahren 1 – 2 und 5 – 6: Unterschied im Distanzmaß Ein Vergleich der unterschiedlichen Distanzen (euklidische oder Poisson) unter Berücksichtigung der funktionalen Form beim Splineclustern (5 – 6) oder ohne Berücksichtigung durch Clustern der Rohdaten (1 – 2) zeigt einen angleichenden Effekt bei den simulierten Daten. Die Abweichungen in der Klassenzuordnung, die in einem unterschiedlichen Distanzmaß begründet sind, erweisen sich als nicht mehr so drastisch, wie bei den realen Daten. Lag vorher der Randindex zwischen Verfahren 1 und 2 bzw. 5 und 6 nie höher als 0.47 (vgl. Tabelle 7.2 auf Seite 50), erreicht er hier sogar Werte bis 0.95.

7.3.1.2. normalverteilt simulierte Daten

Wie verhalten sich nun die einzelnen Partitionierungsmethoden bei klassisch normalverteilten Daten? Eine Verwendung von Clusterverfahren mit der Poissondistanz ist an dieser Stelle nicht sinnvoll und die möglichen Clusterverfahren reduzieren sich auf die Verfahren 1, 3, 4 und 5 (vgl. Box zu den Möglichkeiten zum Clustern auf Seite 32). Die Unterschiede der Verfahren spiegeln damit die Unterschiede in der differenzierten Betrachtungsweise der funktionalen Form der Daten wider. Anhand der auf Basis des Produktes 61 – *Mineralwasser* als normalverteilt simulierte Daten wurden die Lösungen der vier Verfahren auf Wochenebene exemplarisch verglichen. Die Zentren finden sich in Abbildung 7.22, alle weiteren Lösungen für die entsprechend simulierten Daten bei anderen Produkten und beiden Aggregationsstufen lassen sich bei Bedarf Anhang F entnehmen.

In diesem speziellen Fall erkennen alle vier Verfahren ein oberes relativ konstant verlaufendes Cluster bei ca. 2 Einheiten pro Woche. Beim Clustern der Splinekoeffizienten (Verfahren 3) ergibt sich, anders als bei den drei übrigen Verfahren, kein Cluster nahe bei 0, sondern nur eines, dass parallel auf etwas höherem Niveau verläuft. Auffällig sind die fast identisch verlaufenden Lösungen des Splineclusterns (Verfahren 5) und des Clusterns der angepassten Splinewerte (Vefahren 4). Inwieweit das Clustern der Rohdaten den Klassenzuordnungen der anderen Clustermöglichkeiten entspricht, ist vor allem bei Betrachtung des Verlaufs der beiden mittleren Cluster, nicht ganz deutlich. Das Standardverfahren (Ver-

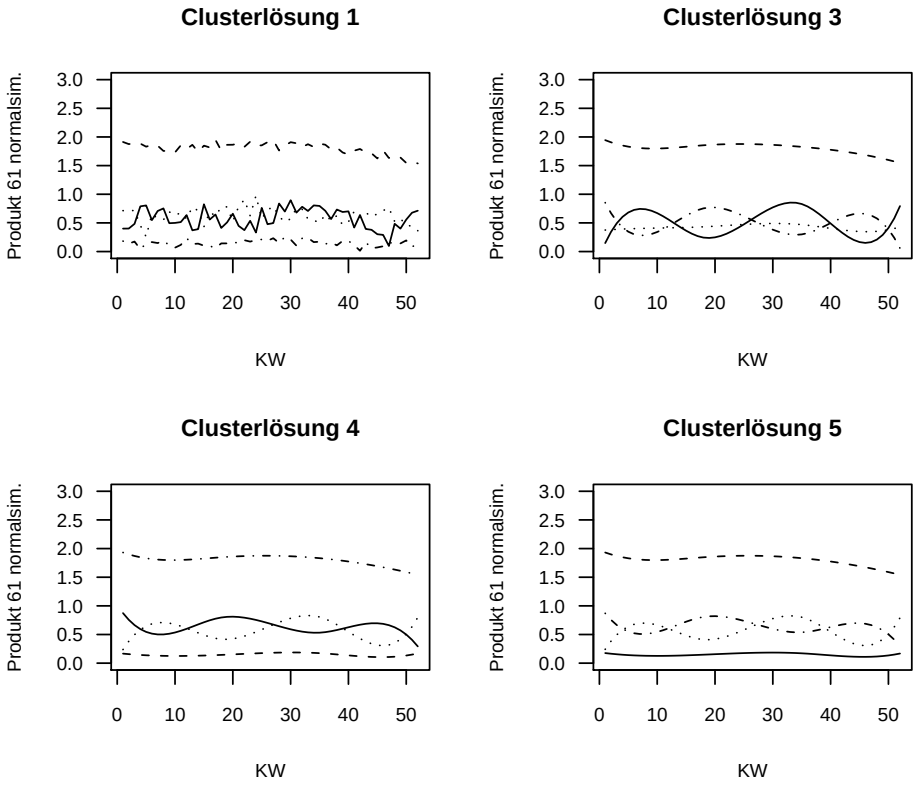


Abbildung 7.22.: Verlauf der Clusterzentren für die einzelnen Clusterlösungen, Produkt 61, normalverteilt simulierte Daten, KW-Ebene

fahren 3) zeigt im Gegensatz zu Verfahren 4 und 5 etwas ausgeprägtere Schwankungen bei diesen Centroiden.

Verfahren 3: Unterschied zum Standardverfahren Ob und wie stark sich Verfahren 3, wie schon bei den poissonverteilten und den realen Daten, bei allen Produkten über beide Aggregationsebenen unterscheidet, soll wiederum anhand der Randindizes zueinander entschieden werden. Diese liegen auf Wochenebene zwischen 0.13 und 0.45, also insgesamt noch etwas niedriger als bei den poissonverteilt simulierten Daten. Dagegen zeigt die monatliche Betrachtung ein durchwachsenes Bild. Hier liegen die Indizes tendenziell hoch. Sie reichen von 0.48 bis 0.92 (vgl. Tabellen im Anhang E). Tabelle 7.6 beinhaltet die Indizes aller Produkte für einen Vergleich zwischen dem Splineclustern und dem Clustern der Rohdaten, sowie zu dem Clustern der angepassten Splinewerte.

Verfahren 1 – 5: Unterschied Clustern der Rohdaten und Splineclustern Bei den vorliegenden Daten sind Verfahren 1 und 5 auf Monatsebene gemessen am Randindex fast identisch, während sich auf Wochenebene niedrigere Ähnlichkeiten abzeichnen. Vor allem bei den strukturarmen Verläufen der Basen von Produkt 63 und 19, die sich auf Wochenebene meist nur zwischen 0 und einer Einheit bewegen, sind die am Randindex bewerteten Unterschiede bei den beiden Verfahren hoch. Die Ähnlichkeit ist bei differenzierteren Transaktionsverläufen dagegen stärker gegeben.

Verfahren 4 – 5: Unterschied Clustern der Splinewerte und Splineclustern Ob es einen Unterschied zwischen Verfahren 4 und 5 gibt, lässt sich ebenfalls nicht so eindeutig beantworten. Bei den normalverteilt simulierten Daten ähneln sich die Clusterzuordnungen manchmal mehr und manchmal weniger. Das durchwachsene Bild hängt von den jeweiligen Ausgangsdaten ab. Oft sind beide Verfahren aber in der Clusterzuordnung so gut wie identisch, wie beispielsweise auf Monatsebene bei den Produkten 63, 61 und 19.

Clusterverfahren i – j	Produkt Aggregationsniveau	63	61	44	19
4 – 5	KW	0.72	0.97	1.00	0.84
	Monat	0.99	1.00	0.63	1.00
1 – 5	KW	0.45	0.71	0.90	0.28
	Monat	0.98	0.99	1.00	0.99

Tabelle 7.6.: Randindex zwischen Clusterverfahren i und Clusterverfahren j aufgeteilt nach Produkt und Aggregationsniveau, normalverteilt simulierte Daten

7.3.2. Vergleich der einzelnen Verfahren hinsichtlich der richtigen Clusterzuordnung

Die Verläufe der Centroide der einzelnen Verfahren lassen sich aufgrund der bekannten Clustercentroide der Simulationsbasis gut beurteilen.

7.3.2.1. poissonverteilt simulierte Daten

Abbildung 7.21 enthält, getrennt für die sechs betrachteten Clustermöglichkeiten, die Verläufe aller Zentren für Produkt 63 – *Eiscreme (Haushaltspackungen)* auf Wochenebene. Ein Vergleich der Zentren mit der Simulationsbasis in Abbildung 7.18 gibt Auskunft über

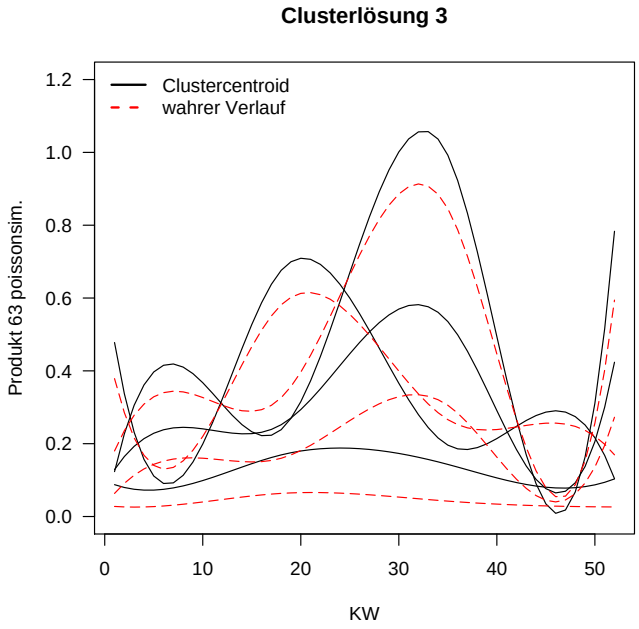


Abbildung 7.23.: Verlauf der Clusterzentren – Clustern der Splinekoeffizienten (B-Splines Grad 3, 2 innere Knoten) mit k-means (k=4) und euklidischer Distanz, poissonverteilt simulierte Daten, Produkt 63, KW-Ebene mit eingezeichneten wahren Verläufen der Simulationsbasis

die Leistungsfähigkeit der entsprechenden Clusterverfahren. Deshalb werden in den Abbildungen 7.23 und 7.24 die wahren Centroidverläufe der Basis und beispielsweise die des Clusterergebnisses des Standardverfahrens bzw. des Splineclusterns mit der Poissondistanz übereinandergelegt.

In der ersten Abbildung ist auffällig, dass die erkannten Clustercentroide nicht optimal die Extrema der wahren Clusterzentren nachzeichnen. Erstere verlaufen leicht vesetzt oberhalb oder auch unterhalb. Das Standardverfahren beim Clustern von Kurven hat bei diesen Daten Schwierigkeiten, die wahren Verläufe aufzudecken. Anders ist dies bei dem zweiten Beispiel, dem Splineclustern mit der Poissondistanz in Verfahren 6. Hier verlaufen die Centroide so gut wie identisch mit den Basiskurven.

Wie gut die Klasseneinteilung der einzelnen Verfahren die wahre Klassenzugehörigkeit trifft, soll anhand des Randindex beurteilt werden. Ist der Randindex hoch, entspricht die vorgeschlagene Partitionierung eher der zugrundeliegenden Einteilung, ist er hingegen niedrig, gelingt es dem Verfahren nicht ausreichend, die entsprechenden Gruppen zuzuweisen. Eine Übersicht über die Ergebnisse ist in Tabelle 7.7 aufgeführt. Bereits aus den vorhergehenden Abschnitten ist bekannt, dass sich das Standardverfahren beim Clustern funktionaler Daten meist von den anderen Clustermöglichkeiten unterscheidet. Anhand der Ergebnisse des Randindex zwischen der wahren Klassenzugehörigkeit und dem Clustern der

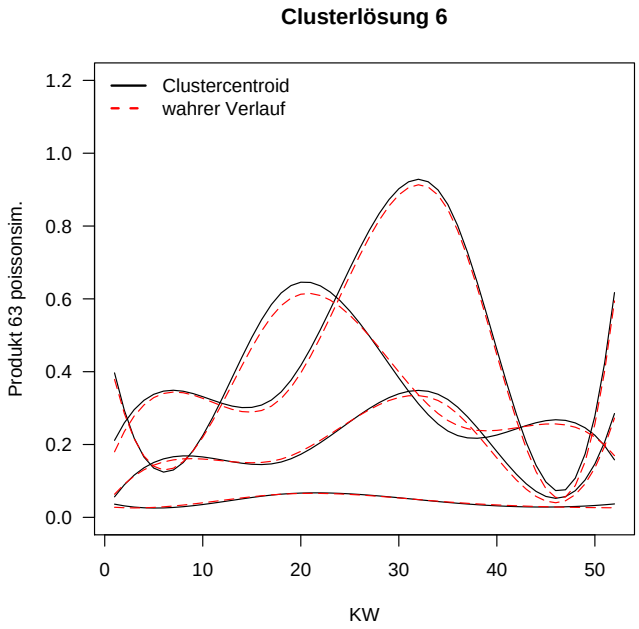


Abbildung 7.24.: Verlauf der Clusterzentren – Clustern der Rohdaten mit Splineclustern (B-Splines Grad 3, 2 innere Knoten, df=6) und Poisson-Distanz (k=4), poissonverteilt simulierte Daten, Produkt 63, KW-Ebene mit eingezeichneten wahren Verläufen der Simulationsbasis

	Verfahren / Aggregationsniveau	**1**	**2**	**3**	**4**	**5**	**6**
Produkt 63	KW	0.70	0.73	0.33	0.76	0.62	0.77
	Monat	0.77	0.78	0.34	0.76	0.76	0.78
Produkt 61	KW	0.52	0.75	0.34	0.59	0.59	0.76
	Monat	0.70	0.83	0.39	0.71	0.72	0.83
Produkt 44	KW	0.84	0.83	0.43	0.85	0.85	0.85
	Monat	0.83	0.63	0.59	0.83	0.83	0.84
Produkt 19	KW	0.47	0.58	0.30	0.50	0.51	0.60
	Monat	0.64	0.74	0.40	0.64	0.64	0.75

Tabelle 7.7.: Randindex zwischen wahrer Klassenzugehörigkeit und den einzelnen Verfahren, poissonverteilt simulierte Daten

Splinekoeffizienten bei den poissonverteilt simulierten Daten wird nun auch deutlich, dass es dem Verfahren am schlechtesten gelingt die echten Klassen aufzudecken. Der Index erreicht maximal den Wert 0.59. Das neue Verfahren des Splineclusterns in Verbindung mit der Poissondistanz zeigt bei den vorliegenden Daten ein gutes Ergebnis. Der Randindex weist bei allen Produkten über beide Aggregationsstufen den höchsten oder zumindest mit den höchsten Wert auf und liegt insgesamt zwischen 0.60 und 0.85.

7.3.2.2. normalverteilt simulierte Daten

Bei den normalverteilt simulierten Kurven schneidet Verfahren 3 auch in den meisten Fällen am schlechtesten ab. So liegt der Randindex auf der Wochenebene bei maximal 0.42. Auf Monatsebene erreicht er dagegen bei dieser Art von Daten weit höhere Werte als bei den poissonverteilt simulierten Daten. Das Analogon zu Verfahren 6 bei poissonverteilt simulierten Beobachtungen ist hier das Splineclustern mit euklidischer Distanz.

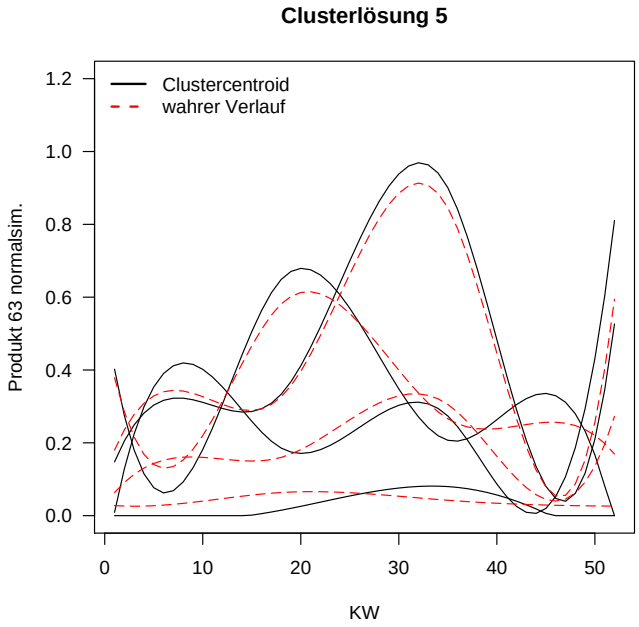


Abbildung 7.25.: Verlauf der Clusterzentren – Clustern der Rohdaten mit Splineclustern (B-Splines Grad 3, 2 innere Knoten, df=6) und euklidischer Distanz (k=4), normalverteilt simulierte Daten, Produkt 63, KW-Ebene mit eingezeichneten wahren Clustercentroiden der Simulationsbasis

Auch hier lässt sich feststellen, dass das Splineclustern die größten oder zumindest gleich große Indizes produziert. Insgesamt ist die Aufdeckung der wahren Cluster auf Kalenderwochenebene eher schlecht. Auf Monatsebene scheinen die Verfahren 1, 4 und 5 in vielen Fällen für diese Daten gleich gut geeignet zu sein.

	Verfahren / Aggregationsniveau	1	2	3	4	5	6
Produkt 63	KW	0.25	-	0.09	0.29	0.30	-
	Monat	0.81	-	0.45	0.82	0.82	-
Produkt 61	KW	0.63	-	0.36	0.66	0.66	-
	Monat	0.87	-	0.75	0.87	0.87	-
Produkt 44	KW	0.72	-	0.42	0.73	0.73	-
	Monat	0.96	-	0.89	0.63	0.96	-
Produkt 19	KW	0.15	-	0.09	0.22	0.22	-
	Monat	0.82	-	0.51	0.82	0.82	-

Tabelle 7.8.: Randindex zwischen wahrer Klassenzugehörigkeit und den einzelnen Verfahren, normalverteilt simulierte Daten

7.4. Vergleich der Clusterlösungen

7.4.1. Vergleich der Ähnlichkeiten

In Abschnitt 7.2.2 und 7.3.1 wurden bereits jeweils die Ähnlichkeiten der einzelnen Verfahren zueinander für die realen und die simulierten Daten besprochen. Um zu sehen, ob gewisse Gemeinsamkeiten zwischen den Clusterlösungen auch über beide Datenlagen (real vs. simuliert) hinweg bestehen, enthält Tabelle 7.2 die Randindizes der Vergleiche zwischen den unterschiedlichen Distanzmaßen (1 – 2 und 5 – 6), sowie den unterschiedlichen funktionalen Betrachtungsweisen (1 – 5, 2 – 6 und 4 – 5).

Die Abbildungen 7.26 und 7.27 enthalten die Boxplots der Randindizes zwischen diesen Verfahren, aufgeteilt nach den zugrundeliegenden Daten. Mit **A** und **B** werden die realen Transaktionsdaten jeweils auf Kalenderwochen- und Monatsebene und mit **C** und **D** bzw. **E** und **F** die entsprechend poissonverteilt bzw. normalverteilt simulierten Daten bezeichnet.

Der Median der Randindizes zwischen dem Clustern der Splinewerte (Verfahren 4) und dem Splineclustern mit euklidischer Distanz (Verfahren 5) ist bei allen Daten am höchsten. Dort besteht demnach offensichtlich der größte Zusammenhang. Auffällig ist zudem, dass die geringe Übereinstimmung in der Klassenzuteilung, gemessen durch den niedrigen Index beim Vergleich von Verfahren mit unterschiedlichen Distanzmaßen (1 – 2 und 5 – 6) bei den realen Daten, bei den poissonsimulierten stark abgeschwächt ist. Zwar ist der Median dieser Indizes weiterhin überall am niedrigsten, aber erreicht dennoch Werte von 0.66 bis 0.84. Beim Vergleich des Clusterns der Rohdaten vs. Splineclustern (1 – 5 und 2 – 6) zeigt sich, dass die Gemeinsamkeiten bei den unterschiedlichen Daten immer recht hoch sind,

		A	B	C	D	E	F
Produkt 63	**1 – 2**	0.41	0.40	0.58	0.73	-	-
	1 – 5	0.90	0.93	0.82	0.97	0.45	0.98
	4 – 5	0.96	0.96	0.97	0.98	0.72	0.99
	2 – 6	0.89	0.93	0.94	0.96	-	-
	5 – 6	0.40	0.40	0.68	0.77	-	-
Produkt 61	**1 – 2**	0.33	0.37	0.79	0.90	-	-
	1 – 5	0.97	0.98	0.63	0.94	0.71	0.99
	4 – 5	0.99	0.98	0.63	0.99	0.97	1.00
	2 – 6	0.82	0.96	0.82	0.97	-	-
	5 – 6	0.37	0.36	0.62	0.90	-	-
Produkt 44	**1 – 2**	0.45	0.47	0.93	0.62		
	1 – 5	0.97	0.98	0.93	0.97	0.90	1.00
	4 – 5	0.98	0.99	0.99	1.00	1.00	0.63
	2 – 6	0.98	0.99	0.92	0.63	-	-
	5 – 6	0.45	0.46	0.95	0.95	-	-
Produkt 19	**1 – 2**	0.26	0.21	0.62	0.72	-	-
	1 – 5	0.54	0.59	0.84	0.97	0.28	0.99
	4 – 5	0.58	0.97	0.97	0.98	0.84	1.00
	2 – 6	0.37	0.49	0.86	0.94	-	-
	5 – 6	0.47	0.41	0.64	0.74	-	-

Tabelle 7.9.: Randindex zwischen verschiedenen Clusterverfahren i – j bei verschiedenen Produkten und Daten;
A: Daten, KW-Ebene
B: Daten, Monatsebene
C: poissonverteilt simulierte Daten, KW-Ebene
D: poissonverteilt simulierte Daten, Monatsebene
E: normalverteilt simulierte Daten, KW-Ebene
F: normalverteilt simulierte Daten, Monatsebene

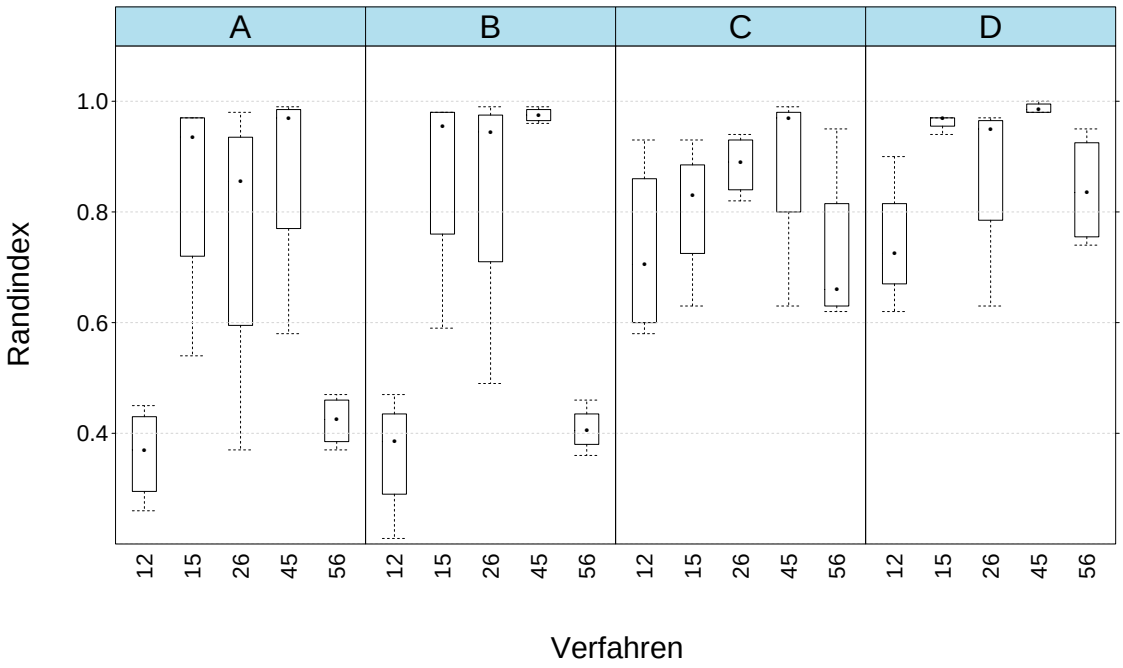


Abbildung 7.26.: Boxplots der Randindizes der Verfahren i und j zueinander, aufgeteilt nach Datensituation A, B, C und D

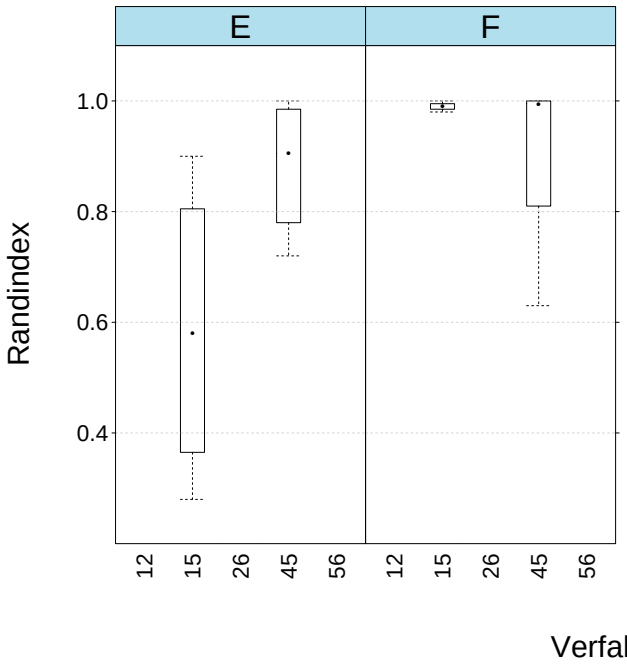


Abbildung 7.27.: Boxplots der Randindizes der Verfahren i und j zueinander, aufgeteilt nach Datensituation E und F

wobei eine Betrachtung auf Monatsebene (Daten **B**, **D** und **E**) zu weit ähnlicheren Klassenzuweisungen kommt, als dies bei einer Betrachtung auf wöchentlicher Basis der Fall ist. Der Vollständigkeit wegen, sei an dieser Stelle noch aufgeführt, dass die Gemeinsamkeiten der Verfahren zum Standardverfahren, dem Clustern der Splinekoeffizienten (Verfahren 3), sowohl bei den realen, als auch den simulierten, i.d.R. am geringsten sind.

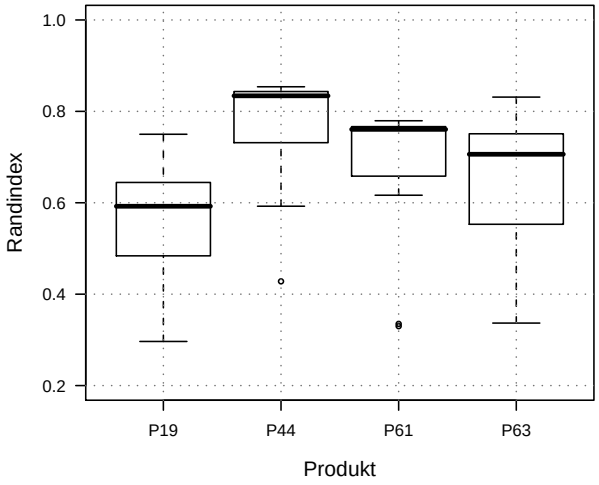
7.4.2. Untersuchung der Leistungsfähigkeit der Verfahren bei poissonverteilt simulierten Daten

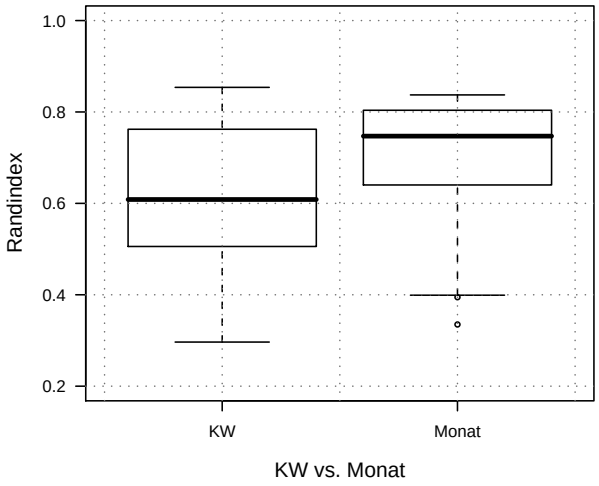
Welches Verfahren scheint nun am geeignetsten zum Clustern von Transaktionsverläufen, als spezieller Fall poissonverteilter Zähldaten? In Abschnitt 7.3.2 wurde bereits kurz auf die Ähnlichkeiten der einzelnen Clusterlösungen mit der wahren Klassenzugehörigkeit bei den simulierten Daten eingegangen. Nun soll abschließend geklärt werden, ob bei den poissonverteilten Daten ein Verfahren zum Clustern funktionaler Daten (signifikant) besser – gemessen am Randindex – ist, als ein anderes. Zudem soll auch kurz festgehalten werden, inwieweit ein unterschiedliches Aggregationsniveau eine Rolle beim Abschneiden der Verfahren spielt. Besonderer Fokus wird darüberhinaus auf das Standardverfahren und das neu entwickelte Splineclustern mit Poissondistanz, als speziell für diese Art von Daten entwickelte Methode, gelegt. Schneidet Verfahren 3, wie in Abschnitt 7.3.2 angedeutet, in jedem Fall schlechter ab, als alle anderen betrachteten Clustermethoden und stellt sich das Splineclustern mit der Poissondistanz als signifikant am besten für die Transaktionsdaten heraus?

In einem ersten Schritt lassen sich Unterschiede in Verfahren, Aggregationsniveau und produktspezifischen Verläufen hinsichtlich des Randindex zur wahren Clusterlösung betrachten (vgl. dazu Tabelle 7.7 auf Seite 64). Dazu sind in der Abbildung 7.28 die Boxplots der Randverteilungen dargestellt.

Das oberste Bild zeigt die Verteilung des Randindex nach den einzelnen produktspezifischen Simulationsbasen. Insgesamt können über alle Verfahren und Aggregationsebenen bei nach Produkt 44 simulierten Daten die besten Ergebnisse erzielt werden, gefolgt von den Verläufen bei Produkt 61 und 63. Am schlechtesten gelingt eine Aufdeckung der wahren Klassen bei Produkt 19, der strukturärmsten Basis. In Abbildung 7.18 auf Seite 54 lassen sich die Simulationsbasen auf Wochenebene noch einmal zum Vergleich betrachten. Wie erwartet, erkennen alle Verfahren gemeinsam betrachtet stärker im Niveau differente Transaktionsverläufe besser, als zwar stark schwankende, aber auf gleichem Niveau verlaufende Transaktionskurven.

Die Unterscheidung hinsichtlich der Aggregationsebene bringt hervor, dass der Randindex bei einer gröberen Betrachtung höher liegt und weniger streut. Dies lässt sich aus Abbildung 7.28 zur Verteilung des Randindex nach KW bzw. Monat im mittleren Bild entnehmen. Der Median des Randindex auf Monatsebene ist hier bei 0.75, während er auf Wochenebene nur bei 0.61 liegt. Eine richtige Klassenzuteilung scheint also durch eine





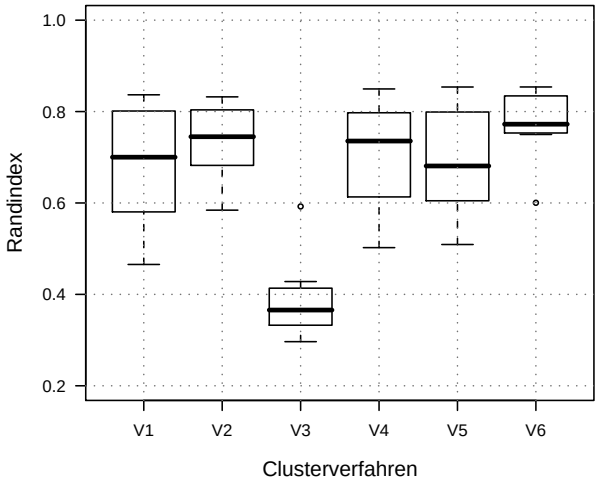


Abbildung 7.28.: Boxplots des Randindex

gröbere Betrachtungsweise begünstigt zu sein.

Aufgeteilt nach den einzelnen Verfahren im untersten Bild wird bestätigt, dass Verfahren 3 wesentlich niedrigere Randindizes hervorbringt, als alle anderen Verfahren. Aus Tabelle 7.10 lässt sich entnehmen, dass der Median des Index hier bei 0.37 liegt, während dieser bei den anderen Verfahren Werte zwischen 0.68 bis hin zu 0.77 beim Splineclustern mit Poissondistanz erreicht. Die Verwendung der Poissondistanz zum Clustern der Rohdaten besitzt den zweithöchsten Medianwert. Diese drei bereits erwähnten Methoden (3, 2 und 6) zeigen einen geringeren Interquartilsabstand und damit eine geringere Streuung im Randindex, als die Verfahren 1, 4 und 5, den Verfahren mit klassischer euklidischer Distanz. Die Methoden mit Poissondistanz liegen demnach bei diesen Daten nicht nur insgesamt höher im Index, sondern erzeugen, über verschiedene Aggregationsniveaus und Transaktionsverläufen betrachtet, gleichmäßigere Ergebnisse.

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
V1	0.466	0.613	0.700	0.683	0.784	0.837
V2	0.584	0.708	0.745	0.735	0.792	0.832
V3	0.296	0.334	0.366	0.389	0.406	0.592
V4	0.502	0.625	0.736	0.706	0.780	0.850
V5	0.509	0.611	0.681	0.692	0.782	0.854
V6	0.600	0.755	0.772	0.772	0.833	0.854

Tabelle 7.10.: Kenngrößen der Randindizes der einzelnen Verfahren

Abbildung 7.29 zeigt eine Aufteilung der Boxplots der Verfahren getrennt nach den beiden zeitlichen Betrachtungsebenen. Auch hier bleiben die grundsätzlichen, das Standardverfahren und die beiden poissondistanzbasierten Verfahren betreffenden Erkenntnisse bestehen: Das Clustern der Splinekoeffizienten bringt die niedrigsten Indizes hervor und die beiden anderen Methoden mit der neuen Distanz die höchsten Medianwerte.

Um diese deskriptiv angelegten Ergebnisse strukturierter zu betrachten, wird eine einfaktorielle Varianzanalyse mit dem Randindex als Zielgröße und den einzelnen Verfahren als faktorielle Einflussgröße gerechnet. Für jede Faktorstufe sind acht gemessene Randindizes verzeichnet, einer für jedes Produkt auf jeder Aggregationsebene. Die einfaktorielle Varianzanalyse setzt voraus, dass die Messungen innerhalb einer Faktorstufe jeweils normalverteilt sind und zudem Varianzhomogenität der Zielgröße zwischen den Faktorstufen besteht. Die erste Voraussetzung wurde mit Hilfe des Kolmogorov-Smirnov-Tests auf Normalverteilung getestet und gilt hier als erfüllt. Auch die zweite Voraussetzung ist nach Überprüfung mit dem Levene-Varianzhomogenitätstest gegeben. Die Varianzanalyse stellt heraus, dass sich mindestens zwei Randindizes hinsichtlich ihres Mittelwertes signifikant unterscheiden. Mit

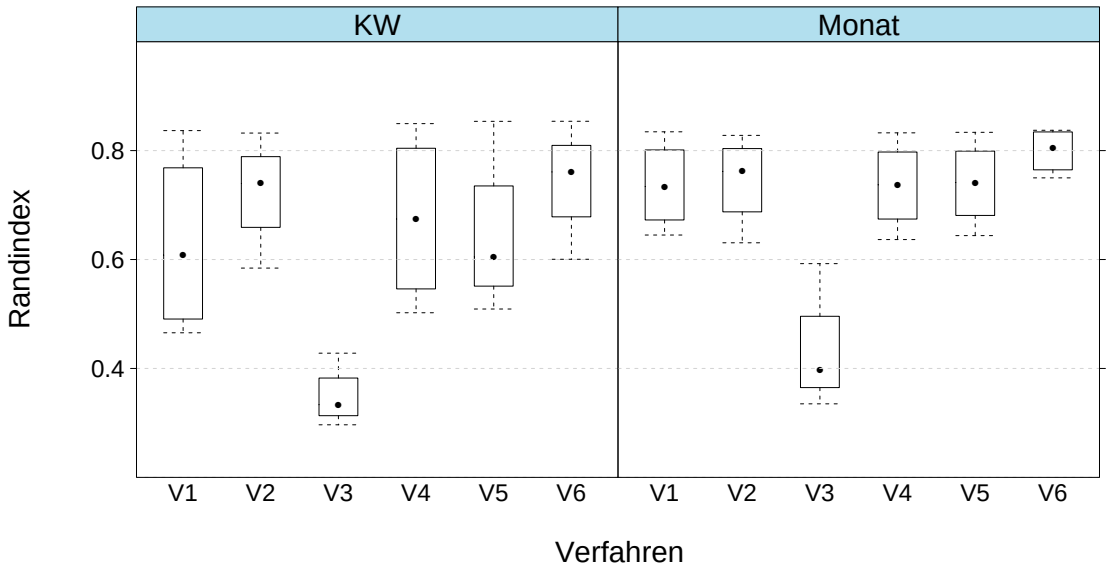


Abbildung 7.29.: Boxplots des Randindex für die unterschiedlichen Verfahren aufgeteilt nach Aggregationsniveaus

Hilfe linearer Kontraste sollen nun die bestehenden Hypothesen zu den einzelnen Clustermethoden überprüft werden. Unterscheiden sich die Verfahren mit der Poissondistanz (2 und 6) signifikant von denen mit der euklidischen Distanz (1, 4 und 5)? Gibt es einen Unterschied zwischen der funktionalen Betrachtungsweise mittels Splineclustern (5 und 6) und dem Clustern der Rohdaten (1 und 2)? Bringt das Splineclustern gegenüber dem Clustern der Rohdaten unter Verwendung der Poisson-Distanz einen signifikanten Vorteil? Zeigt Verfahren 6 gegenüber 1, 2, 4 und 5 eine höhere Leistung in der Klassenzuordnung und ist das Standardverfahren bei dieser Datenlage signifikant schlechter als alle anderen Verfahren?

	Estimate	p-value
Verfahren V2,V6 vs V1,V4,V5	0.36	0.10
Verfahren V5,V6 vs V1,V2	0.09	0.56
Verfahren V2 vs. V6	-0.09	0.11
Verfahren V1,V2,V4,V5 vs. V6	-0.27	0.12
Verfahren V1,V2,V4,V5,V6 vs. V3	1.64	0.00

Tabelle 7.11.: Test linearer Kontraste

Bei linearen Kontrasten ist die Nullhypothese, dass der durchschnittliche Mittelwert von bestimmten Gruppen dem durchschnittlichen Mittelwert bestimmter anderer Gruppen ent-

spricht. In Tabelle 7.11 sind die Ergebnisse der Tests mit dem jeweiligen p-value aufgeführt. Es lässt sich entnehmen, dass die poissondistanzbasierten Verfahren einen höheren mittleren Index besitzen, aber dieser Effekt nicht signifikant ist. Ein Vergleich der funktionalen Betrachtungsweise mit dem Clustern der Rohdaten über beide Distanzmaße bzw. nur mit dem Poisson-Distanzmaß ergibt einen positiven Effekt für die funktionale Herangehensweise, dieser ist aber ebenfalls nicht signifikant. Für das neue Verfahren 6 kann auch kein signifikanter Unterschied zu den anderen Methoden festgestellt werden, wenn auch insgesamt ein höherer mittlerer Index verzeichnet wird. Die einzig gesicherte Erkenntnis ist, dass für diese simulierten Daten das Standardverfahren, das Clustern der Splinekoeffizienten mit euklidischer Distanz, signifikant schlechter abschneidet als die übrigen Methoden.

Eine weitere Möglichkeit einen post-hoc-Test zur Analyse der Mittelwertsdifferenzen, der das multiple Signifikanzniveau hält, durchzuführen, ist der multiple Test nach Tukey (*Tukey´s honestly significant difference (HSD)*).[19] Dabei werden alle paarweisen Differenzen miteinander verglichen. Das Ergebnis ist in Abbildung 7.30 dargestellt.

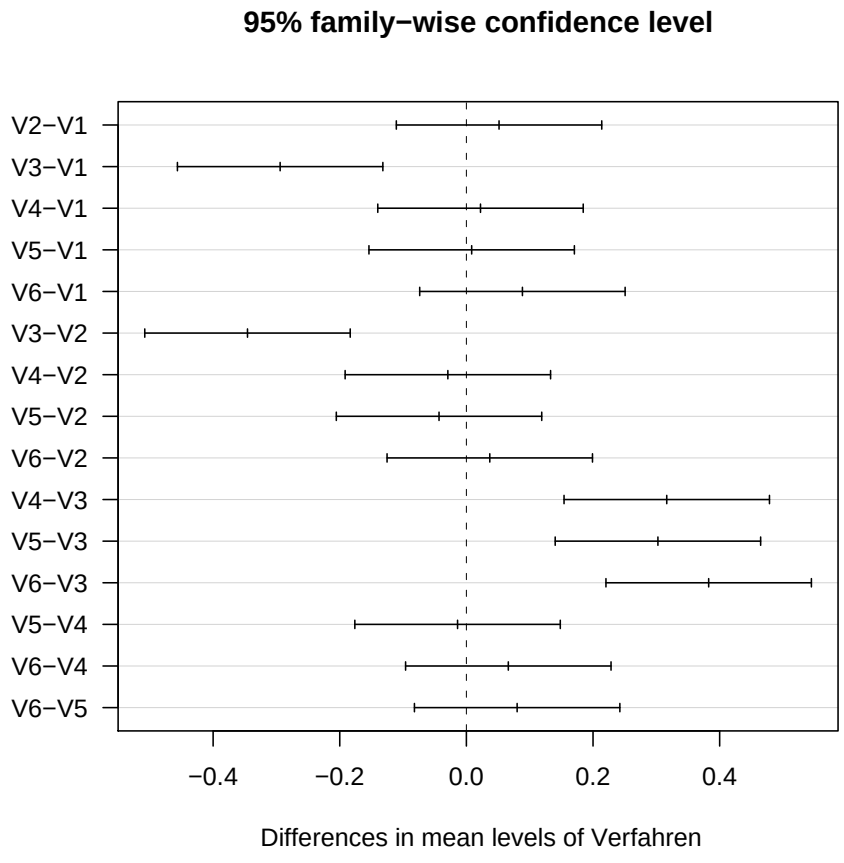


Abbildung 7.30.: multipler Test nach TukeyHSD

[19] vgl. Toutenburg (1994) und Everitt and Hothorn (2010).

Ist in dieser Abbildung die Null in dem 95%−Intervall ($0 \in KI$) eingeschlossen, so liegt kein signifikanter Unterschied zwischen den Mittelwerten der zwei miteinander verglichenen Verfahren vor. Lediglich Verfahren 3 zeigt mit den anderen Verfahren einen signifikanten Unterschied im mittleren Randindex. Bei Verfahren 6 ist der Randindex zwar im Mittel immer höher als bei den anderen Verfahren, dabei aber nicht signifikant. Darüberhinaus zeigt auch das Clustern der Rohdaten mit der Poisson-Distanz (Verfahren 2), bis auf einen Vergleich mit dem Splineclustern bei gleichem Distanzmaß, eine mittlere positive Auswirkung, dabei aber auch nicht signifikant.

KAPITEL 8

Fazit und Ausblick

Hauptziel der vorliegenden Arbeit war es, einen Überblick über bestehende Verfahren funktionaler Clusteranalyse zu geben, sowie mögliche Erweiterungen und Abänderungen des etablierten Verfahrens – Clustern von Splinekoeffizienten mit dem k-means-Algorithmus – im Rahmen von Transaktionsdaten zu diskutieren. Die funktionale Clusteranalyse lässt sich, wie üblich, in modellbasierte und heuristische Ansätze trennen. Innerhalb dieser Aufteilung existieren wiederum verschiedene Möglichkeiten, glatte Kurven zur funktionalen Betrachtung an die diskret gemessenen Daten anzupassen. Der klassische, heuristische, partitionierende Ansatz des k-means Clusterns der Koeffizienten einer Splineexpansion verwendet als Distanzmaß die euklidische Distanz, während der verallgemeinerte k-means-Algorithmus andere denkbare Distanzen zulässt. Aufgrund einer Betrachtung von Transaktionsverläufen, die eine Folge von Zähldaten darstellen, erschien es naheliegend, diese besondere Datenlage im Clusteralgorithmus zu berücksichtigen. Durch eine Erweiterung des k-means-Algorithmus mit der neu entwickelten Poisson-Distanz (vgl. Abschnitt 6.1) wurde dies erreicht. Zudem wurde in Abschnitt 6.2 eine alternative Herangehensweise gegenüber dem Standardverfahren zur Beachtung der Funktionalität von Verläufen vorgestellt. Beim *Splineclustern* werden die Centroid-Vektoren als funktional betrachtet und im Rahmen des k-means-Algorithmus jeweils geglättet, nicht die zugrundeliegenden Daten selbst.
Auf Basis echter Transaktionsverlaufsdaten über Einkäufe von Haushalten, sowie bei auf deren Grundlage simulierten Daten, wurden in Abschnitt 7 die Ähnlichkeiten bestimmter Verfahren und deren Leistungsfähigkeit hinsichtlich der richtigen Klassenzuteilung untersucht. Die Überprüfung der Ähnlichkeiten erbrachte über die realen und simulierten Daten nur beschränkt einheitliche Ergebnisse. Offensichtlich ist, dass sich das Standardverfahren sowohl stark vom Clustern der Rohdaten, als auch vom Splineclustern unterscheidet. Die größten Gemeinsamkeiten, gemessen am Median des Randindex zwischen den Verfahren,

bestehen zwischen dem Clustern der Splinewerte an den Erhebungszeitpunkten (Verfahren 4) und dem Splineclustern (Verfahren 5), jeweils mit euklidischer Distanz. Darüberhinaus sind die Ähnlichkeiten zwischen Verfahren mit gleichem Distanzmaß aber unterschiedlicher funktionaler Betrachtungsweise (1 – 5, 2 – 6) i.d.R. höher, als bei einem unterschiedlichen Distanzmaß und gleicher Vorgehensweise (1 – 2, 5 – 6).

Für den speziellen Fall von poissonverteilt simulierten Transaktionsdaten ließ sich feststellen, dass das Standardverfahren wesentlich schlechter bzgl. der richtigen Klassenzuordnung abschnitt, als die anderen Verfahren. Außerdem schien in diesem Fall ein angemessen gewähltes Distanzmaß mehr zur Aufdeckung der wahren Cluster beizutragen, als eine funktionale Methodik wie das Splineclustern. Offen für weitere Untersuchungen bleibt demnach, ob das Splineclustern bei anderer Datenlage mehr Gewinn bringt. Dabei kann auch darüber diskutiert werden, inwieweit eine funktionale Herangehensweise überhaupt Vorteile gegenüber einem Clustern der Rohdaten bringt. Zumindest für die vorliegenden poissonverteilt simulierten Daten war der Randindex mit Poissondistanz, also die Kombination des angemessenen Distanzmaßes und einer funktionalen Clustermethode, wertmäßig immer am höchsten. Ob dies auch generell für andere Zähldaten gilt, wäre ebenfalls noch zu überprüfen.

Abschließend sollte auch noch erwähnt werden, dass die Idee des Splineclusterns, anders als beispielsweise das klassische Verfahren, auf gleiche Messzeitpunkte beschränkt ist. Hier ließe sich noch überlegen, inwieweit dieses Verfahren hinsichtlich ungleicher Anzahl und nicht identischer Messzeitpunkte ausgeweitet werden kann.

Das Splineclustern und die Poissondistanz zeigen, dass eine den Daten angemessene Clustersystematik sehr bedeutend für die Qualität der Ergebnisse ist. Wie beim verallgemeinerten k-means-Algorithmus sollte auch beim funktionalen Clustern immer die Datenlage in die Entscheidung über ein geeignetes Distanzmaß miteinfließen. Die Berücksichtigung einer Funktionalität in den Daten lässt sich auf unterschiedlichste Weise erreichen. Ob das Splineclustern, als eine dieser Möglichkeiten, im Einzelfall bessere Ergebnisse liefert, ließ sich an dieser Stelle noch nicht eindeutig klären.

Literaturverzeichnis

Abraham, C., P. Cornillon, E. Matzner-Løber, and N. Molinari (2003). Unsupervised curve clustering using b-splines. *The Scandinavian Journal of Statistics 30*, 581–595.

Arabie, L. H. . P. (1985). Comparing partitions. *Journal of Classification 2*(1), 193–218.

Banfield, J. D. and A. E. Raftery (1993, September). Model-based gaussian and non-gaussian clustering. *Biometrics 49*, 803–821.

Chiou, J.-M. and P.-L. Li (2007). Functional clustering and identifying substructures of longitudinal data. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 69*(4), 679–699.

Chiou, J.-M. and P.-L. Li (2008, December). Correlation-based functional clustering via subspace projection. *Journal of the American Statistical Association 103*(484), 1684–1692.

Cuesta-Albertos, J., A. Gordaliza, and C. Matrán (1997). Trimmed k-means: An attempt to robustify quantizers. *The Annals of Statistics 25*, 553–576.

Dempster, A. P., N. M. Laird, and D. B. Rubin (1977). Maximum likelihood for incomplete data via the em algorithm. *Journal of the Royal Statistical Society, Series B 39*, 1–38.

Everitt, B. S. and T. Hothorn (2010). *A Handbook of Statistical Analyses Using R* (2 ed.). Boca Raton: Chapman & Hall/CRC.

Fahrmeir, L. and C. Heumann (WS 2008/09). Skript zur Vorlesung Schätzen & Testen I.

Fahrmeir, L., T. Kneib, and S. Lang (2009). *Regression : Modelle, Methoden und Anwendungen* (2 ed.). Berlin, Heidelberg: Springer.

Flury, B. D. (1990). Principal points. *Biometrika 77*(1), 33–41.

Flury, B. D. (1993). Estimation of principle points. *Applied Statistics 42*(1), 139–151.

Flury, B. D. and T. Tarpey (1993, November). Representing a large collection of curves: A case for principal points. *The American Statistician 47*(4), 304–306.

Fraley, C. and A. E. Raftery (2002). model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association 97*(458), 611–631.

García-Escudero, L., A. Gordaliza, and C. Matrán (2003). Trimming tools in exploratory data analysis. *Journal of Computational and Graphical Statistics 12*, 434–449.

García-Escudero, L. A. and A. Gordaliza (2005). A proposal for robust curve clustering. *Journal of Classification 22*, 185–201.

Hartigan, J. A. and M. A. Wong (1979). A K-means clustering algorithm. *Applied Statistics 28*, 100–108.

Hastie, T., R. Tibshirani, and J. Friedman (2001). *The Elements of Statistical Learning: Data Mining, Inference and Prediction*. Springer Series in Statistics. New York: Springer.

Hitchcock, D. B., J. G. Booth, and G. Casella (2007, December). The effect of pre-smoothing functional data on cluster analysis. *Journal of Statistical Computation and Simulation 77*(12), 1043–1055.

James, G. M. and C. A. Sugar (2003). Clustering for sparsely sampled functional data. *Journal of the American Statistical Association 98*, 397–408.

Jank, W. and G. Shmueli (2006). Studying heterogeneity of price evolution in ebay auctions via functional clustering. In *Handbook of Information Systems Series: Business Computing*. Adomavicius and Gupta.

Laird, N. M. and J. H. Ware (1982). Random-effects models for longitudinal data. *Biometrics 38*, 963–974.

Leisch, F. (2006). A toolbox for k-centroids cluster analysis. *Computational Statistics & Data Analysis 51*, 526–544.

Luan, Y. and H. Li (2003). Clustering of time-course gene expression data using a mixed-effects model with b-splines. *Bioinformatics 19*(4), 474–482.

Luan, Y. and H. Li (2004). Model-based methods for identifying periodically expressed genes based on time course microarray gene expression data. *Bioinformatics 20*(3), 332–339.

Ma, P., C. I. Castillo-Davis, W. Zhong, and J. S. Liu (2006). A data-driven clustering method for time course gene expression data. *Nucleic Acids Research 34*(4), 1261–1269.

Ramsay, J. O. and B. W. Silverman (2002). *Applied Functional Data Analysis - Methods and Case Studies*. Springer Series in Statistics. New York: Springer.

Ramsay, J. O. and B. W. Silverman (2006). *Functional Data Analysis* (Second Edition ed.). Springer Series in Statistics. New York: Springer.

Rand, W. M. (1971, Dezember). Objective criteria for the evaluation of clustering methods. *Journal or the American Statistical Association 66*(336), 846–850.

Rossi, F., B. Conan-Guez, and A. E. Golli (2004). Clustering functional data with the som algorithm. *ESANN proceedings - European Symposium on Artificial Neural Networks*, 305–312.

Serban, N. and L. Wasserman (2005, September). Cats: Clustering after transformation and smoothing. *Journal of the American Statistical Association 100*(471), 990–999.

Shimizu, N. and M. Mizuta (2008). Functional principal points and functional cluster analysis. In *Computational Intelligence Paradigms*. Lakhmi C. Jain and Mika Sato-Ilic and Maria Virvou and George A. Tsihrintzis and Valentina Emilia Balas and Canicious Abeynayake.

Spanos, R. G. G. . P. D. (1991). *Stochastic Finite Elements: A Spectral Approach*. New York: Springer.

Steinley, D. (2006). K-means clustering: A half-century synthesis. *British Journal of Mathematical and Statistical Psychology 59*(1), 1–34.

Tarpey, T. (2007a, February). Linear transformations and the k-means clustering algorithm: Applications to clustering curves. *The American Statistician 61*(1), 34–40.

Tarpey, T. (2007b). A parametric k-means algorithm. *Computational Statistics 22*, 71–89.

Tarpey, T. and K. K. Kinateder (2003). Clustering functional data. *Journal of Classification 20*, 93–114.

Tarpey, T., E. Petkova, and R. T. Ogden (2003, December). Profiling placebo responders by self-consistent partitioning of functional data. *Journal of the American Statistical Association 98*(464), 850–858.

Toutenburg, H. (1994). *Versuchsplanung und Modellwahl*. Heidelberg: Physica-Verlag.

Yeung, K. Y., C. Fraley, A. Murua, A. E. Raftery, and W. L. Ruzzo (2001). Model-based clustering and data transformations for gene expression data. *Bioinformatics 17*(10), 977–987.

Yeung, K. Y. and W. L. Ruzzo (2001). Principal components analysis for clustering gene expression data. *Bioinformatics 17*(9), 763–774.

ANHANG A

Verlauf der Clusterzentren, Transaktionsdaten

A.1. Kalenderwochenebene

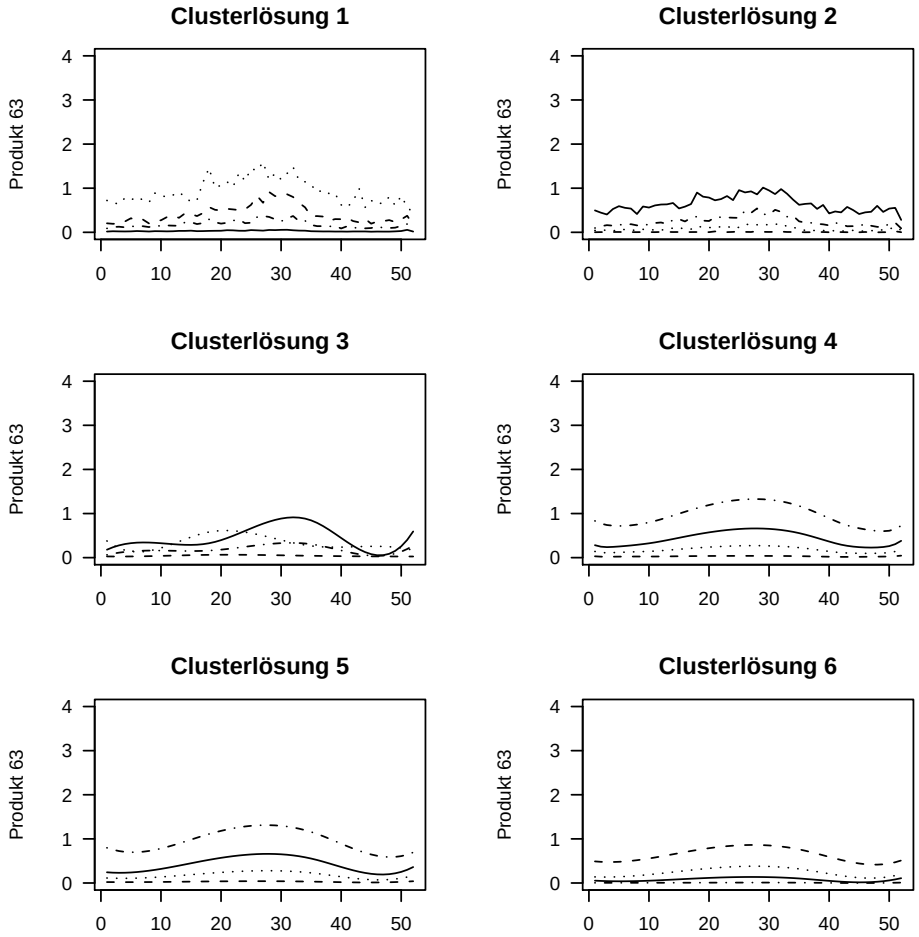


Abbildung A.1.: Verlauf der Clusterzentren der einzelnen Clusterlösungen, Produkt 63, KW-Ebene

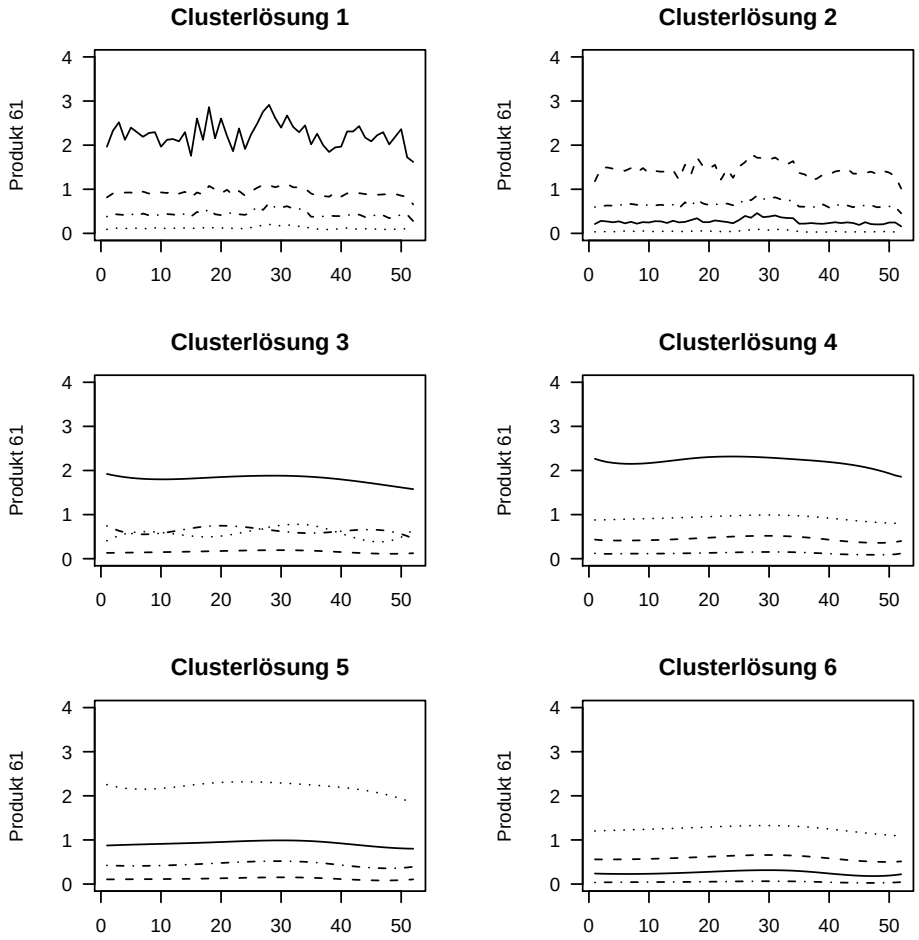


Abbildung A.2.: Verlauf der Clusterzentren der einzelnen Clusterlösungen, Produkt 61, KW-Ebene

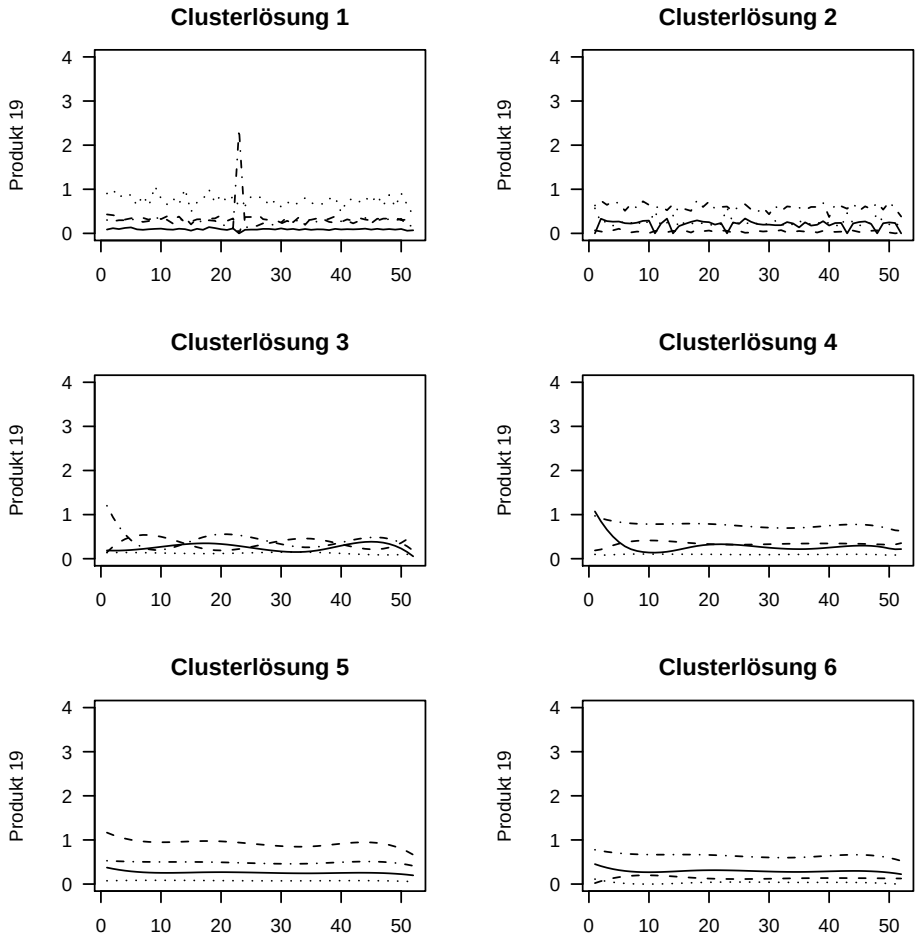


Abbildung A.3.: Verlauf der Clusterzentren der einzelnen Clusterlösungen, Produkt 19, KW-Ebene

A.2. Monatsebene

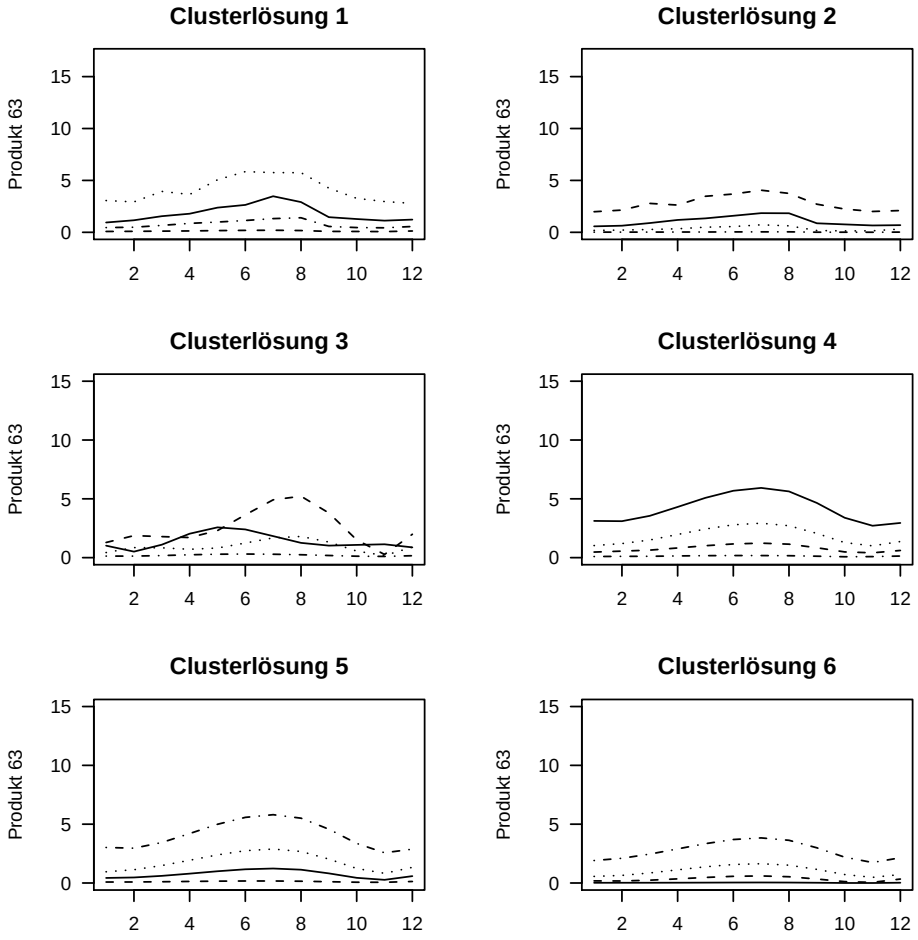


Abbildung A.4.: Verlauf der Clusterzentren der einzelnen Clusterlösungen, Produkt 63, Monatsebene

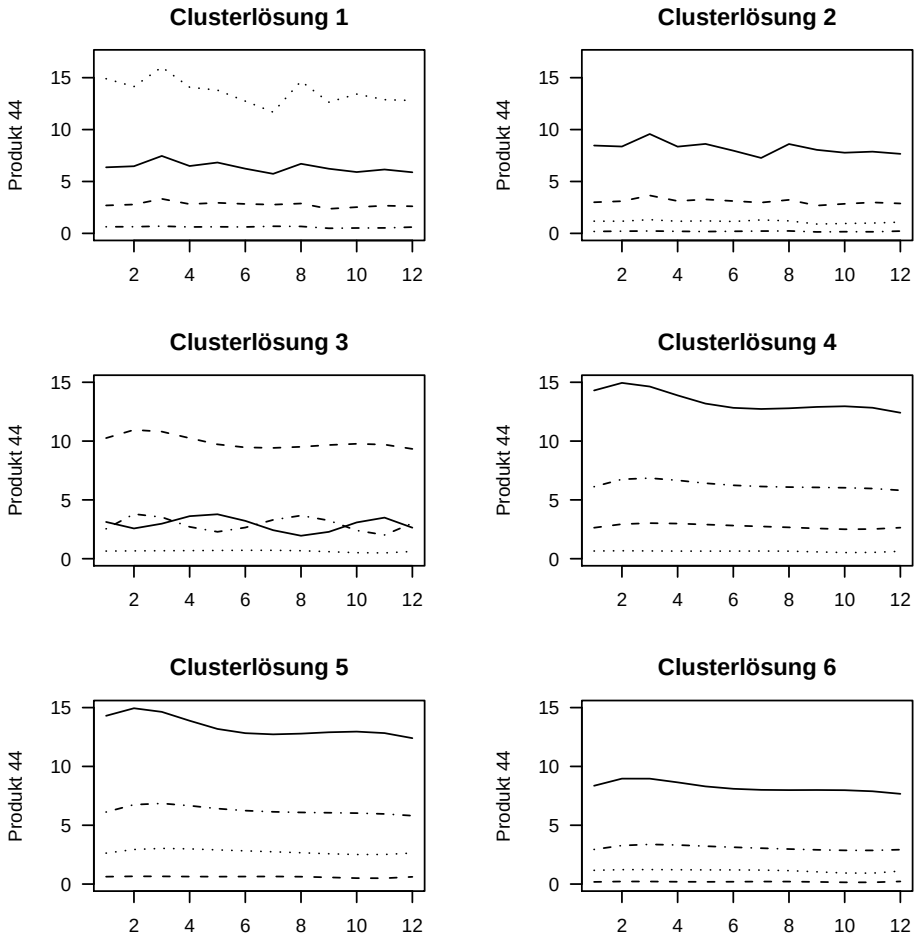


Abbildung A.5.: Verlauf der Clusterzentren der einzelnen Clusterlösungen, Produkt 44, Monatsebene

ANHANG B

Randindex der Clusterlösungen zueinander, Transaktionsdaten

B.1. Kalenderwochenebene

	1	2	3	4	5	6
Rohdaten, k-means, eukl - 1	1.00	0.33	0.48	0.96	0.97	0.37
Rohdaten, k-means, poi - 2	0.33	1.00	0.32	0.33	0.33	0.82
Splinekoeffizienten, k-means, eukl - 3	0.48	0.32	1.00	0.49	0.48	0.32
Splinewerte, k-means, eukl - 4	0.96	0.33	0.49	1.00	0.99	0.37
Rohdaten, Splineclustern, eukl - 5	0.97	0.33	0.48	0.99	1.00	0.37
Rohdaten, Splineclustern, poi - 6	0.37	0.82	0.32	0.37	0.37	1.00

Tabelle B.1.: Vergleich der Clusterlösungen, Produkt 61, KW-Ebene

	1	2	3	4	5	6
Rohdaten, k-means, eukl - 1	1.00	0.45	0.56	0.97	0.97	0.46
Rohdaten, k-means, poi - 2	0.45	1.00	0.30	0.45	0.45	0.98
Splinekoeffizienten, k-means, eukl - 3	0.56	0.30	1.00	0.56	0.56	0.30
Splinewerte, k-means, eukl - 4	0.97	0.45	0.56	1.00	0.98	0.46
Rohdaten, Splineclustern, eukl - 5	0.97	0.45	0.56	0.98	1.00	0.45
Rohdaten, Splineclustern, poi - 6	0.46	0.98	0.30	0.46	0.45	1.00

Tabelle B.2.: Vergleich der Clusterlösungen, Produkt 44, KW-Ebene

	1	2	3	4	5	6
Rohdaten, k-means, eukl - 1	1.00	0.26	0.18	0.65	0.54	0.41
Rohdaten, k-means, poi - 2	0.26	1.00	0.12	0.26	0.39	0.37
Splinekoeffizienten, k-means, eukl - 3	0.18	0.12	1.00	0.24	0.17	0.09
Splinewerte, k-means, eukl - 4	0.65	0.26	0.24	1.00	0.58	0.42
Rohdaten, Splineclustern, eukl - 5	0.54	0.39	0.17	0.58	1.00	0.47
Rohdaten, Splineclustern, poi - 6	0.41	0.37	0.09	0.42	0.47	1.00

Tabelle B.3.: Vergleich der Clusterlösungen, Produkt 19, KW-Ebene

B.2. Monatsebene

	1	2	3	4	5	6
Rohdaten, k-means, eukl - 1	1.00	0.40	0.50	0.93	0.93	0.40
Rohdaten, k-means, poi - 2	0.40	1.00	0.27	0.40	0.40	0.93
Splinekoeffizienten, k-means, eukl - 3	0.50	0.27	1.00	0.49	0.49	0.26
Splinewerte, k-means, eukl - 4	0.93	0.40	0.49	1.00	0.96	0.41
Rohdaten, Splineclustern, eukl - 5	0.93	0.40	0.49	0.96	1.00	0.40
Rohdaten, Splineclustern, poi - 6	0.40	0.93	0.26	0.41	0.40	1.00

Tabelle B.4.: Vergleich der Clusterlösungen, Produkt 63, Monatsebene

	1	2	3	4	5	6
Rohdaten, k-means, eukl - 1	1.00	0.37	0.51	0.98	0.98	0.37
Rohdaten, k-means, poi - 2	0.37	1.00	0.33	0.37	0.36	0.96
Splinekoeffizienten, k-means, eukl - 3	0.51	0.33	1.00	0.50	0.50	0.33
Splinewerte, k-means, eukl - 4	0.98	0.37	0.50	1.00	0.98	0.37
Rohdaten, Splineclustern, eukl - 5	0.98	0.36	0.50	0.98	1.00	0.36
Rohdaten, Splineclustern, poi - 6	0.37	0.96	0.33	0.37	0.36	1.00

Tabelle B.5.: Vergleich der Clusterlösungen, Produkt 61, Monatsebene

	1	2	3	4	5	6
Rohdaten, k-means, eukl - 1	1.00	0.47	0.58	0.98	0.98	0.47
Rohdaten, k-means, poi - 2	0.47	1.00	0.31	0.46	0.46	0.99
Splinekoeffizienten, k-means, eukl - 3	0.58	0.31	1.00	0.58	0.58	0.31
Splinewerte, k-means, eukl - 4	0.98	0.46	0.58	1.00	0.99	0.46
Rohdaten, Splineclustern, eukl - 5	0.98	0.46	0.58	0.99	1.00	0.46
Rohdaten, Splineclustern, poi - 6	0.47	0.99	0.31	0.46	0.46	1.00

Tabelle B.6.: Vergleich der Clusterlösungen, Produkt 44, Monatsebene

	1	2	3	4	5	6
Rohdaten, k-means, eukl - 1	1.00	0.21	0.27	0.60	0.59	0.26
Rohdaten, k-means, poi - 2	0.21	1.00	0.15	0.22	0.22	0.49
Splinekoeffizienten, k-means, eukl - 3	0.27	0.15	1.00	0.28	0.27	0.17
Splinewerte, k-means, eukl - 4	0.60	0.22	0.28	1.00	0.97	0.40
Rohdaten, Splineclustern, eukl - 5	0.59	0.22	0.27	0.97	1.00	0.41
Rohdaten, Splineclustern, poi - 6	0.26	0.49	0.17	0.40	0.41	1.00

Tabelle B.7.: Vergleich der Clusterlösungen, Produkt 19, Monatsebene

ANHANG C

Verlauf der Clusterzentren der Basen der simulierten Daten

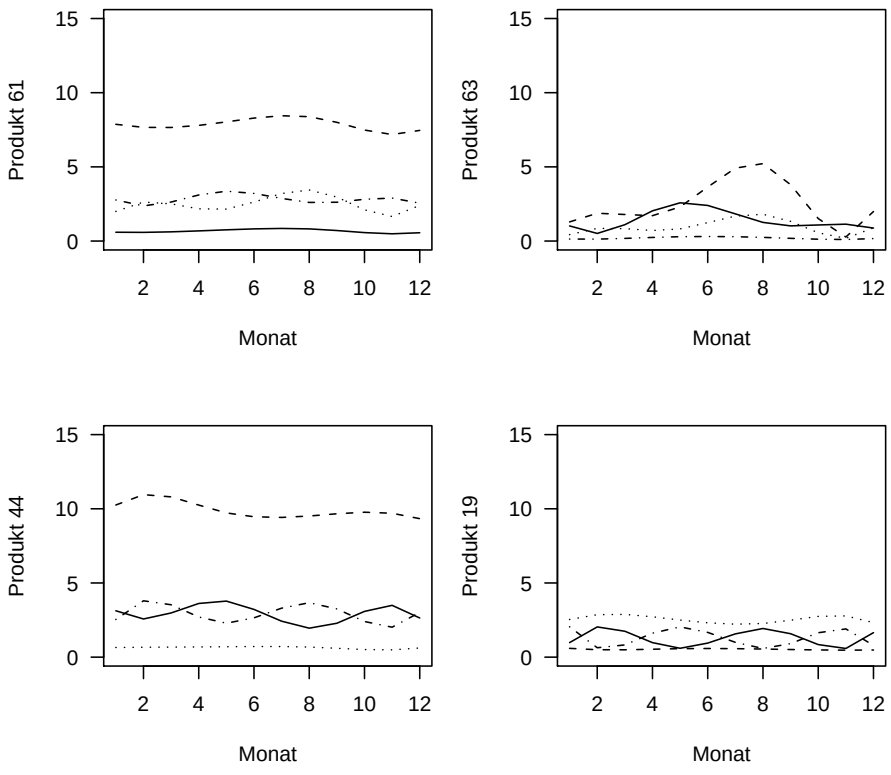


Abbildung C.1.: Verlauf der Clusterzentren der als Basis dienenden Cluster für die simulierten Daten, Monatsebene

ANHANG D

Verlauf der Clusterzentren, poissonverteilt simulierte Daten

D.1. Kalenderwochenebene

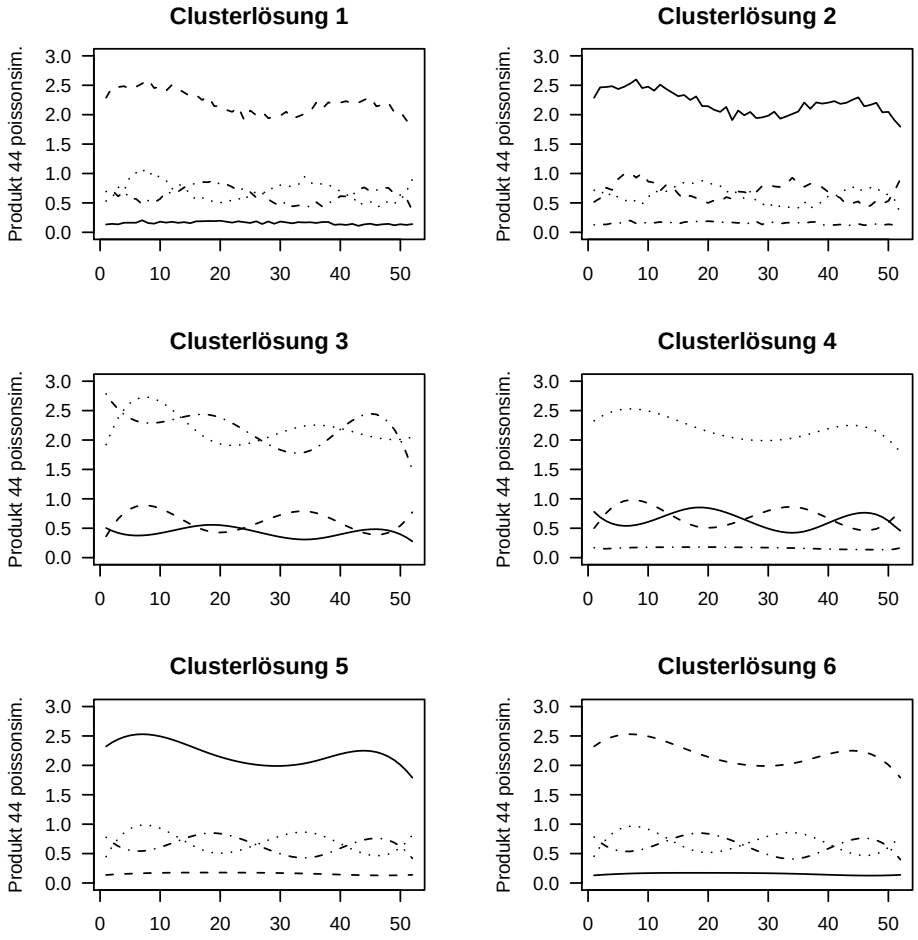


Abbildung D.1.: Verlauf der Clusterzentren der einzelnen Clusterlösungen, Produkt 44, poissonsimulierte Daten, KW-Ebene

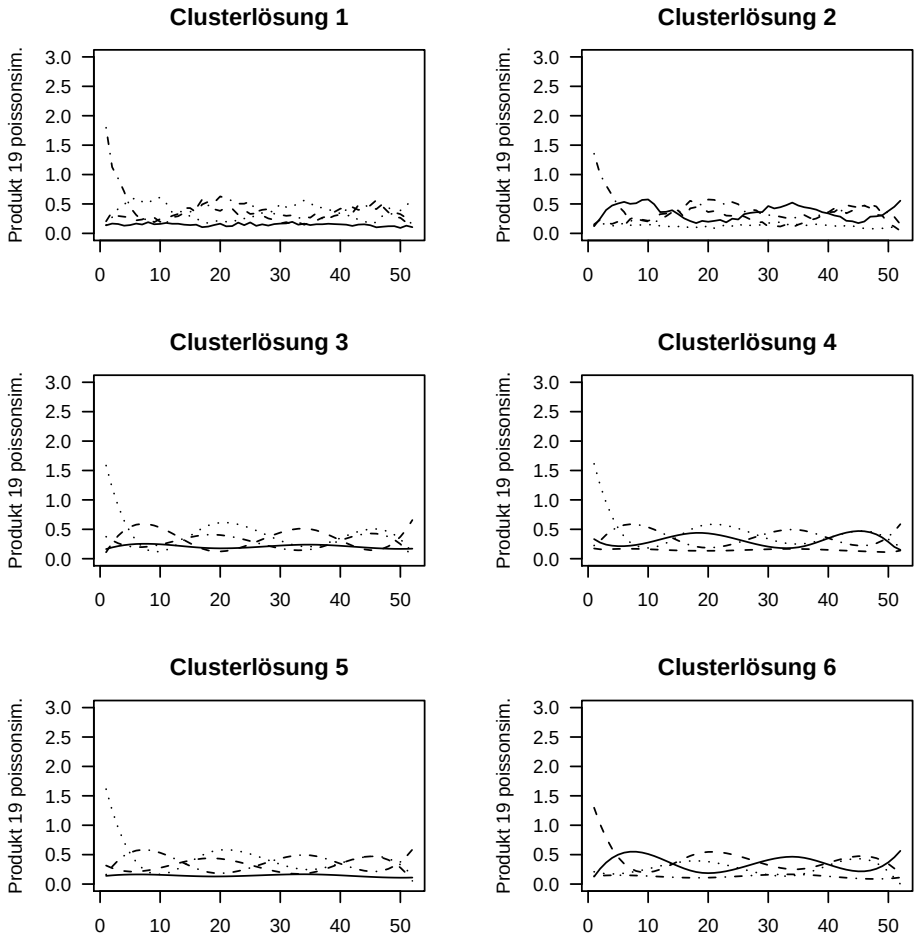


Abbildung D.2.: Verlauf der Clusterzentren der einzelnen Clusterlösungen, Produkt 19, poissonsimulierte Daten, KW-Ebene

D.2. Monatsebene

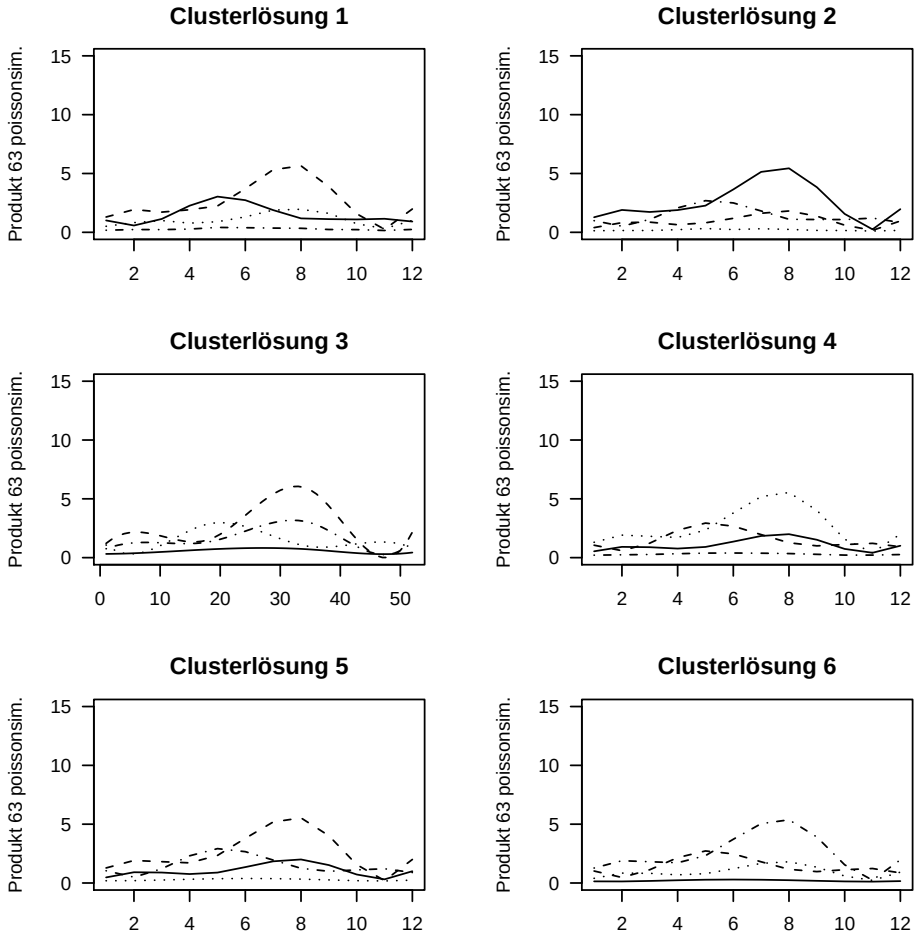


Abbildung D.3.: Verlauf der Clusterzentren der einzelnen Clusterlösungen, Produkt 63, poissonsimulierte Daten, Monatsebene

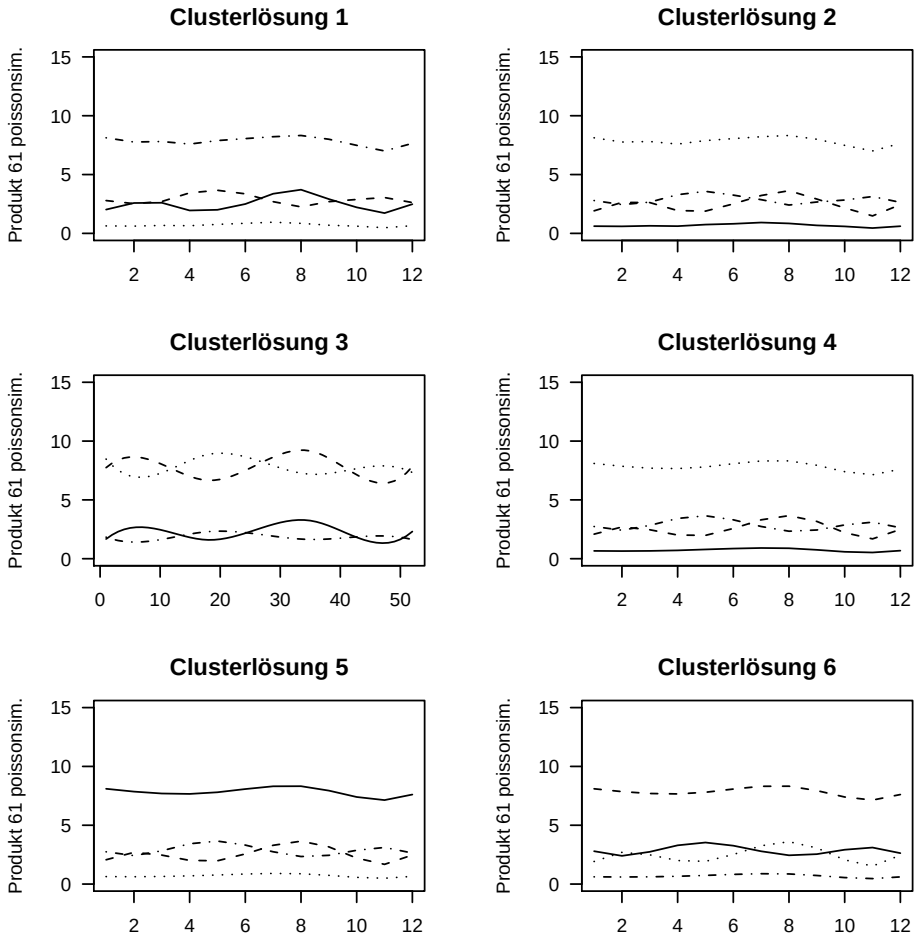


Abbildung D.4.: Verlauf der Clusterzentren der einzelnen Clusterlösungen, Produkt 61, poissonsimulierte Daten, Monatsebene

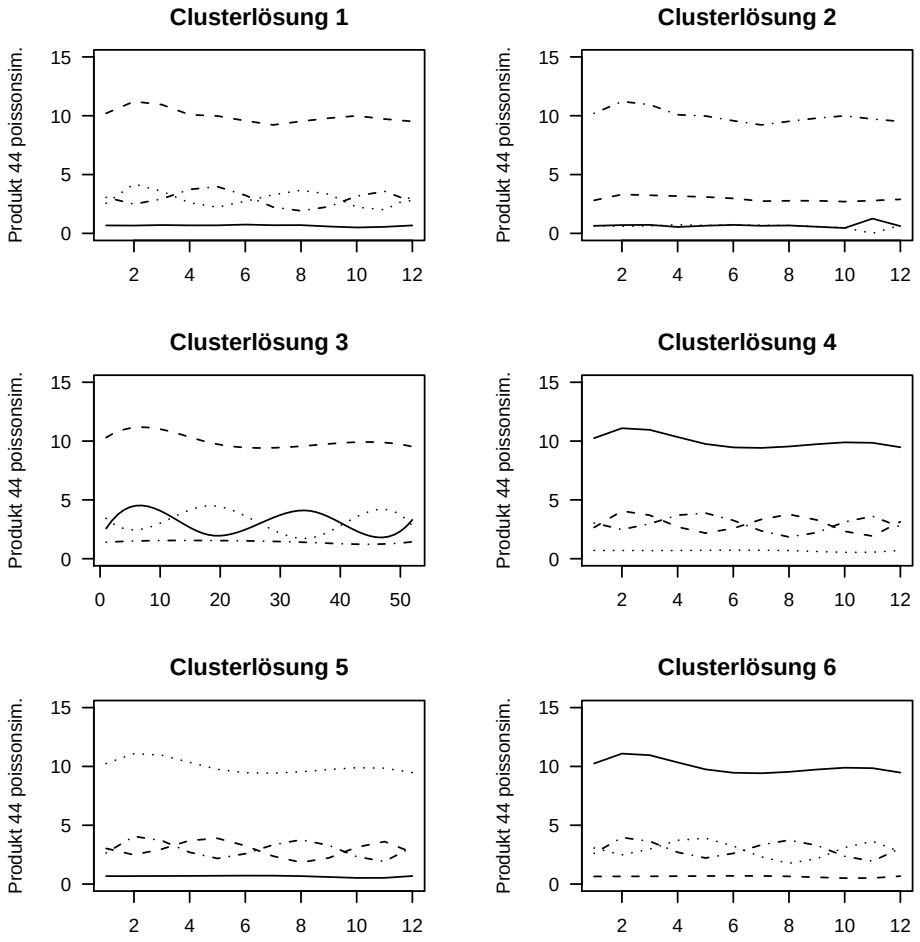


Abbildung D.5.: Verlauf der Clusterzentren der einzelnen Clusterlösungen, Produkt 44, poissonsimulierte Daten, Monatsebene

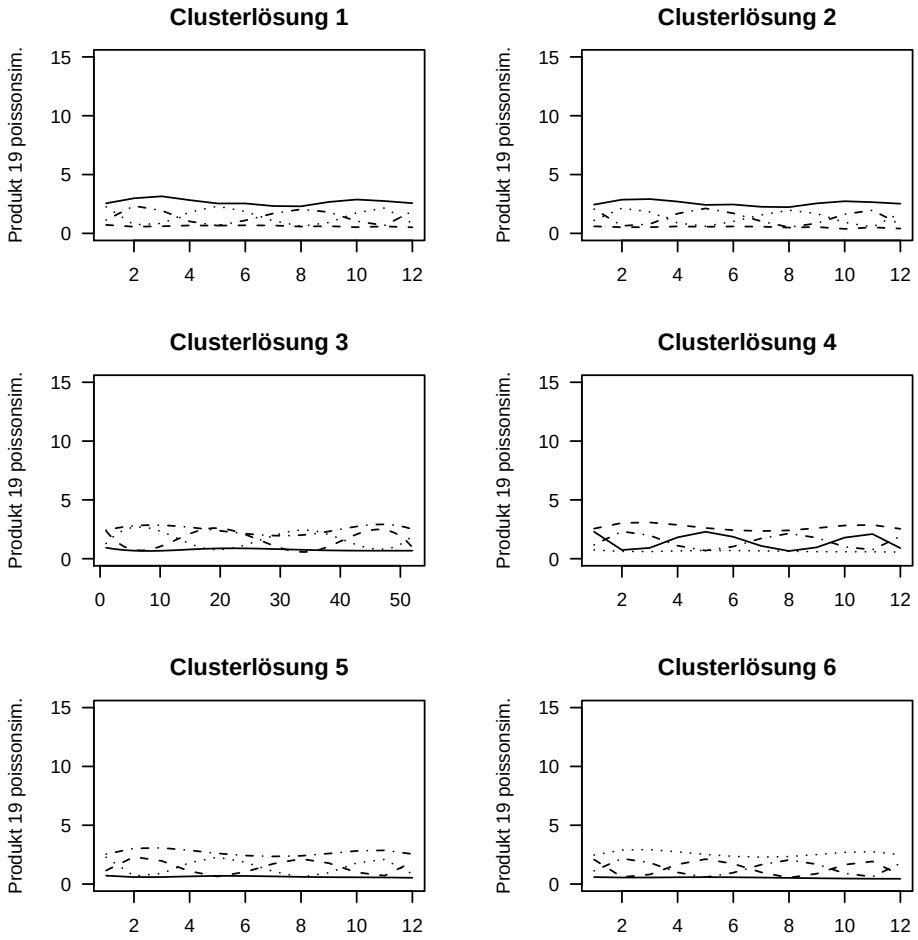


Abbildung D.6.: Verlauf der Clusterzentren der einzelnen Clusterlösungen, Produkt 19, poissonsimulierte Daten, Monatsebene

ANHANG E

Randindex der Clusterlösungen zueinander

	Daten, KW-Ebene	Daten, Monatsebene	poissonsim. Daten, KW-Ebene	poissonsim. Daten, Monatsebene	normalsim. Daten, KW-Ebene	normalsim. Daten, Monatsebene
1-2	0.41	0.40	0.58	0.73	-	-
1-3	0.48	0.50	0.45	0.46	0.13	0.48
1-4	0.88	0.93	0.82	0.96	0.46	0.99
1-5	0.90	0.93	0.82	0.97	0.45	0.98
1-6	0.40	0.40	0.59	0.75	-	-
2-3	0.30	0.27	0.38	0.41	-	-
2-4	0.42	0.40	0.66	0.74	-	-
2-5	0.41	0.40	0.67	0.75	-	-
2-6	0.89	0.93	0.94	0.96	-	-
3-4	0.48	0.49	0.44	0.46	0.17	0.48
3-5	0.48	0.49	0.44	0.46	0.16	0.48
3-6	0.28	0.26	0.38	0.42	-	-
4-5	0.96	0.96	0.97	0.98	0.72	0.99
4-6	0.40	0.41	0.68	0.75	-	-
5-6	0.40	0.40	0.68	0.77	-	-

Tabelle E.1.: Randindizes der einzelnen Verfahren zueinander aufgeteilt nach den zugrundeliegenden Daten, Produkt 63

	Daten, KW-Ebene	Daten, Monatsebene	poissonsim. Daten, KW-Ebene	poissonsim. Daten, Monatsebene	normalsim. Daten, KW-Ebene	normalsim. Daten, Monatsebene
1-2	0.33	0.37	0.79	0.90	-	-
1-3	0.48	0.51	0.34	0.38	0.37	0.80
1-4	0.96	0.98	0.76	0.93	0.71	0.99
1-5	0.97	0.98	0.63	0.94	0.71	0.99
1-6	0.37	0.37	0.75	0.89	-	-
2-3	0.32	0.33	0.34	0.39	-	-
2-4	0.33	0.37	0.81	0.89	-	-
2-5	0.33	0.36	0.62	0.89	-	-
2-6	0.82	0.96	0.82	0.97	-	-
3-4	0.49	0.50	0.39	0.39	0.43	0.80
3-5	0.48	0.50	0.31	0.38	0.42	0.80
3-6	0.32	0.33	0.40	0.39	-	-
4-5	0.99	0.98	0.63	0.99	0.97	1.00
4-6	0.37	0.37	0.92	0.90	-	-
5-6	0.37	0.36	0.62	0.90	-	-

Tabelle E.2.: Randindizes der einzelnen Verfahren zueinander aufgeteilt nach den zugrundeliegenden Daten, Produkt 61

	Daten, KW-Ebene	Daten, Monatsebene	poissonsim. Daten, KW-Ebene	poissonsim. Daten, Monatsebene	normalsim. Daten, KW-Ebene	normalsim. Daten, Monatsebene
1-2	0.45	0.47	0.93	0.62		
1-3	0.56	0.58	0.49	0.62	0.44	0.92
1-4	0.97	0.98	0.94	0.97	0.90	0.63
1-5	0.97	0.98	0.93	0.97	0.90	1.00
1-6	0.46	0.47	0.91	0.93	-	-
2-3	0.30	0.31	0.48	0.35	-	-
2-4	0.45	0.46	0.90	0.62	-	-
2-5	0.45	0.46	0.90	0.62	-	-
2-6	0.98	0.99	0.92	0.63	-	-
3-4	0.56	0.58	0.52	0.62	0.45	0.59
3-5	0.56	0.58	0.52	0.62	0.45	0.92
3-6	0.30	0.31	0.50	0.62	-	-
4-5	0.98	0.99	0.99	1.00	1.00	0.63
4-6	0.46	0.46	0.95	0.95	-	-
5-6	0.45	0.46	0.95	0.95	-	-

Tabelle E.3.: Randindizes der einzelnen Verfahren zueinander aufgeteilt nach den zugrundeliegenden Daten, Produkt 44

	Daten, KW-Ebene	Daten, Monatsebene	poissonsim. Daten, KW-Ebene	poissonsim. Daten, Monatsebene	normalsim. Daten, KW-Ebene	normalsim. Daten, Monatsebene
1-2	0.26	0.21	0.62	0.72	-	-
1-3	0.18	0.27	0.40	0.51	0.13	0.53
1-4	0.65	0.60	0.84	0.96	0.29	0.99
1-5	0.54	0.59	0.84	0.97	0.28	0.99
1-6	0.41	0.26	0.59	0.75	-	-
2-3	0.12	0.15	0.37	0.44	-	-
2-4	0.26	0.22	0.65	0.69	-	-
2-5	0.39	0.22	0.67	0.71	-	-
2-6	0.37	0.49	0.86	0.94	-	-
3-4	0.24	0.28	0.42	0.51	0.19	0.54
3-5	0.17	0.27	0.43	0.51	0.19	0.54
3-6	0.09	0.17	0.36	0.45	-	-
4-5	0.58	0.97	0.97	0.98	0.84	1.00
4-6	0.42	0.40	0.63	0.72	-	-
5-6	0.47	0.41	0.64	0.74	-	-

Tabelle E.4.: Randindizes der einzelnen Verfahren zueinander aufgeteilt nach den zugrundeliegenden Daten, Produkt 19

ANHANG F

Elektronischer Anhang

elektronischer Anhang auf beigelegter CD-Rom

Eigenständigkeitserklärung

Hiermit versichere ich, dass ich die vorliegende Master Thesis selbstständig und ohne Benutzung anderer als die angegebenen Hilfsmittel angefertigt habe.

München, den 25. Januar 2010

Birgit Oellinger